WITHIN THE SHADOWS

The visual world of
Shadows of Doubt

COLE JEFFERIES | MILES ATKINSON

SHADOWS OF DOUBT

Welcome to this inside look at how we built the visual world of Shadows of Doubt. In this book we tell the story about how the visual style of Shadows came about, and as that world is so intertwined with the project's code and development history, it makes sense to take you on a journey through the project's timeline and showcase the process from start to finish.

Throughout you'll be taken on a journey from idea inception up to our current release; we kept an extensive development blog throughout which recorded the thought process on all kinds of design decisions; so I'll be revisiting many of them while keeping the main focus on the visuals. These development blogs are later supplemented by artwork development pieces from our lead voxel artist Miles Atkinson, who developed most of the visuals for the game.

Naturally, putting this all together has been a bit of a nostalgic trip back through the years so I felt it fit to comment on my past development diaries from the perspective of the present. I hope you enjoy this journey as much as we've enjoyed developing the game!

Cole Jefferies

Cole Jefferies, Project Lead.

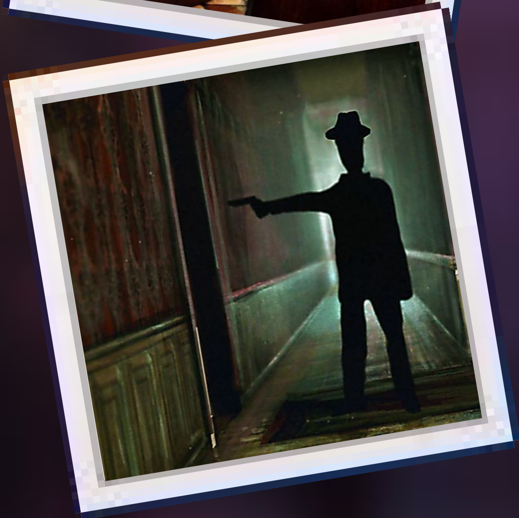
PRE-PLANNING

Although the game didn't really form until very late 2016 or early 2017 (depending on how you really mark the 'start' of the project), the DNA for the kind of atmosphere present in the game today was something of a running theme throughout my interests. Before my programming days I enrolled in an art focused education; completing a degree in illustration before venturing into the world as a want-to-be illustrator, or animator depending on which day you asked me. Throughout this time my go-to art was already embedded in a world which may seem a little familiar; at its core, cityscapes that blended the new with the old to create a kind of alternate timeline world. Often moody. Often raining.

These pieces were largely driven by influences such as Blade-Runner (1982), Bioshock (2007) and Metropolis (1927). Very much gateway influences to the kind of late-teen/early twenties steampunk'y sci-fi mood which captivated me.

It wasn't long before my passion for visuals and generally attempting to make cool stuff crossed paths with my passion for gaming and the indie game boom of the early 2010s. This is where I started my game dev career; progressing through several small scale iOS titles with mixed success, then followed by a kickstarted city/deck builder Concrete Jungle in 2015.

The moderate success of Concrete Jungle, I figured at the time, gave me just about enough runway to take a proper stab at a dream project. In this case, that project was a vast, simulated game set in a semi-futuristic noir world. This was really the start of what was to become Shadows of Doubt.



EARLY DAYS OF DEVELOPMENT

Shadows of Doubt is sometimes billed as a solo developer project, which isn't true as later in the project saw major contributions from a small but talented team of Miles Atkinson (voxels, more from him later!) Stark Holborn (writer), Nick Dymond (audio), multiple other programmers Austin Holland, Josh Regan, Karl Sandegren among many other talented contractors. But when we start the story, it was just me on a modest budget with a vague idea of what I wanted to achieve, so lets pick up the story here...

SHADOWS OF DOUBT
DEVELOPER PROFILE

Cole Jefferies

Lead Developer
Shadows of Doubt

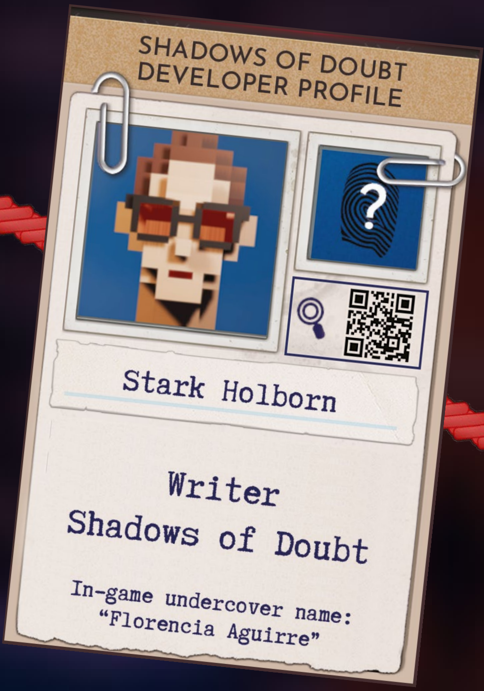
In-game undercover name:
"Shan Dai"

SHADOWS OF DOUBT
DEVELOPER PROFILE

Miles Atkinson

Voxel Artist
Shadows of Doubt

In-game undercover name:
"Jude Hurd"



SHADOWS OF DOUBT
DEVELOPER PROFILE

Stark Holborn

Writer
Shadows of Doubt

In-game undercover name:
"Florencia Aguirre"

SHADOWS OF DOUBT
DEVELOPER PROFILE

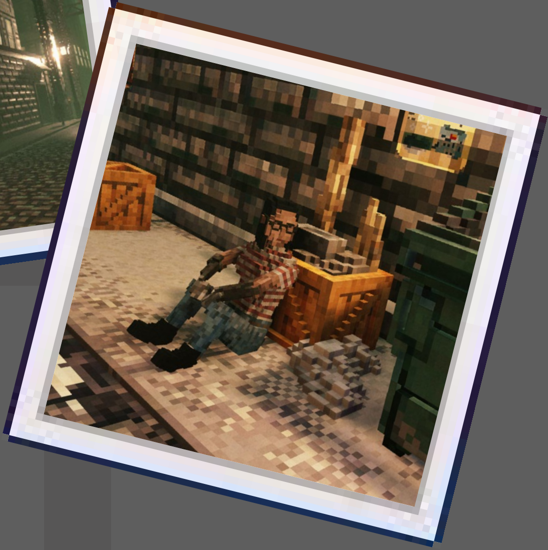
Nick Dymond

Audio Designer
Shadows of Doubt

In-game undercover name:
"Sebastian Childress"

Building a World

Development blogs with updates and commentary by Cole Jefferies



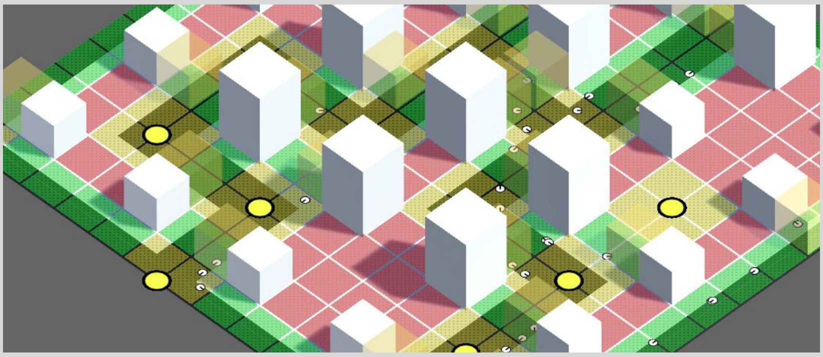
The first development blog begins in 2018, introduced as Shadows of Doubt: A Detective Management game. By this point the game had already been in development for well over a year and I was slowly piecing together the core ideas in search of an elusive vertical slice to pitch to publishers. Little did I know at this point just how long a project this was to become. Some of the featureset described below is long forgotten: This phase of development saw the game morph into something largely different from what is described here, but you can still see a lot of the initial DNA that made it through.

This was/is really the game's core idea that made it through to the Shadows of today. Most of the rest of these first couple of years were spent figuring out how best to build a game around it.

April 20th, 2018

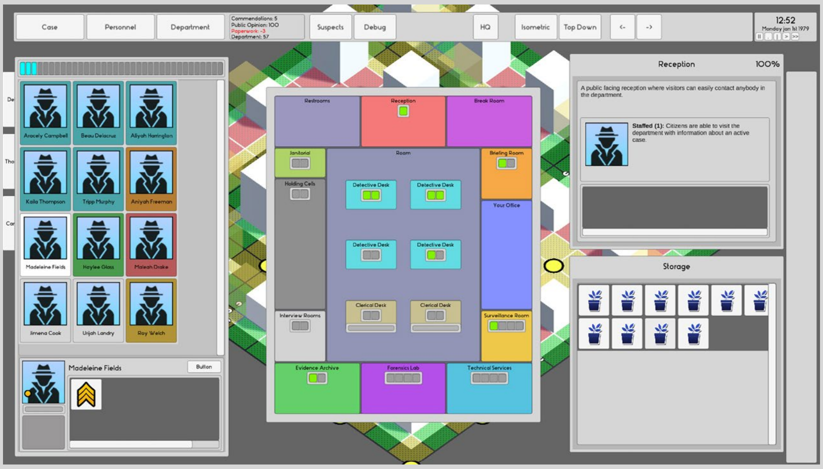
Hello everybody, I'm writing today to 'unveil' what I've been working on after Concrete Jungle. I write that in quotations because there's really not much to look at yet. I've been working on a detective management game called Shadows of Doubt in which you deploy your agents to solve murder cases.

The city is procedurally generated, but not as you know it. Every street, every business, every citizen is different and fully simulated. They're not just the mindless wanderers you find in most open world or large-scale games, nor are they static characters with a few canned lines to say. They have jobs, favourite places to eat and things to do, friends, relationships, enemies, and unique fingerprints. Within the myriad of simulated citizens is a killer- it's your department's job to find them and bring them to justice! What the game is essentially, is a very complex social simulation that happens behind the scenes, and an intuitive set of detective tools that lets the player interact with it.



My ultimate goal with Shadows of Doubt is to create a really engrossing, involving mystery game using a management-style progression system as a hook, as well as the emergent narrative of the procedural cases.

The management mechanics lend well to this. Instead of searching crime scenes yourself, you're a notch above; making important decisions like which suspects to question or shadow, which buildings to stake out and where to best spend this week's department budget. As you acquire more evidence, you'll piece together cases, eventually leading to arrests, and hopefully putting that killer away for good- and ranking up your agents in the process.



Commendations: 5
Public Opinion: 100
Paperwork: -3
Department: 57

A public facing reception where visitors can easily contact anybody in the department.

Staffed (1): Citizens are able to visit the department with information about an active case.

I maintain that this idea could make a cool game. What I didn't realise at this point, was that my initial vision for what I wanted to make didn't gel that well with the management genre.

Your main enemy in the quest for 'whodunnit' is time; just like in real cases the first 24 hours is usually the most important. The evidence is constantly degrading, citizens are forgetting who they saw and when. Everything is becoming fuzzier- hindering your chances of acquiring incriminating evidence. You can pause the game at any point using the space bar, but orders and research will require in-game time. This keeps the focus on your decision making, ratcheting up suspense while giving the player as much time as they want to make important decisions

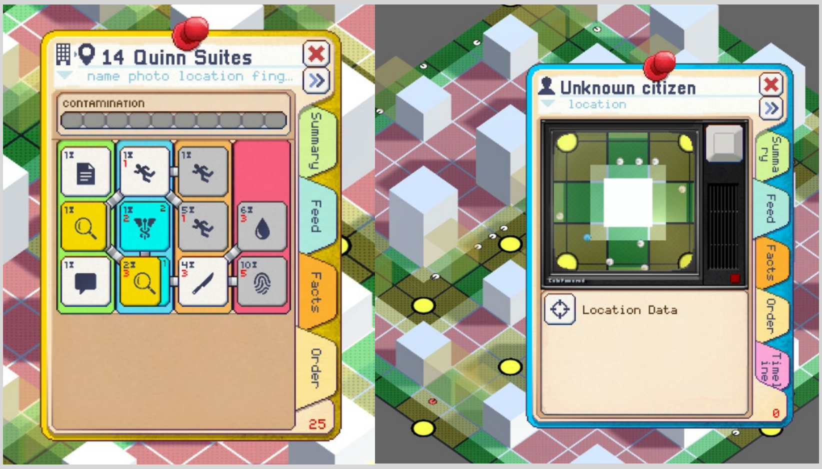
Pressure from upstairs will keep you on your toes. Progress on cases will bring rewards, such as cadets fresh from the academy or new upgrades for your department.

I share my first thoughts on interface and graphics at this point; still well aware that this is early stage artwork, it's nevertheless interesting to see the decision point to use voxels; which would go on to shape the graphical identity of the game.

June 23rd, 2018

For the art style, I'm going with pixel art for 2D interface graphics, mixed with voxel style 3D buildings and 2D sprite characters. I've always been fond of the pixel art style, the interface reminds me of the management genre's golden age in the 90s (memories of Bullfrog games spring to mind). One fear, however, is that it's not an ideal fit for the amount of information I need to display on screen- it's a fine balance I'm going to have to get right.

This is my first time designing pixel art UI, and adjusting to the size/scaling restrictions has taken some patience!



It quickly became clear to me that the interaction with the city shouldn't just be a mechanism for passing the time but a core part of the game play.

I've been mulling over a new fascinating inclusion to the game. A lot of the game involves time passing, although an essential resource, there are periods where there's not really much for the player to do (while waiting on those lab results, etc). This is typical of the genre and isn't really an issue because similar games will just speed up time by several factors and pause when something interesting happens (e.g. XCOM). This is what happens in Shadows of Doubt at the moment, and it's absolutely fine. BUT… Currently, there are isometric and top-down view options only. I'm going to have to make the game pretty anyway, and the 3D city is right there, with all the citizens in their simulated glory. Why not just let the player walk the streets in those phases of 'time passing'? All it would need really is a basic FPS controller. I think being able to walk the streets could add a real connection to the city and its population.

How cool would it be to spend your time waiting reading the morning's newspaper in the greasy spoon across the street over coffee? Or staring out your rainy office window overlooking the city while you wait for your agents to perform an interrogation of the prime suspect. There's something irresistibly noir about that, and I'm going to have to pursue it to some degree. Especially since the game already requires a lot of those assets to be suitably detailed and constructed in 3D anyway. I'm not talking copious amounts of detail here by the way. What I'm picturing is noir pixel art style 3D similar to explore 'em up game Bernband.

And so, the idea starts to form into what you see today! A rabbit hole indeed; an idea which extended the project about four years but transformed the game into what it is now.

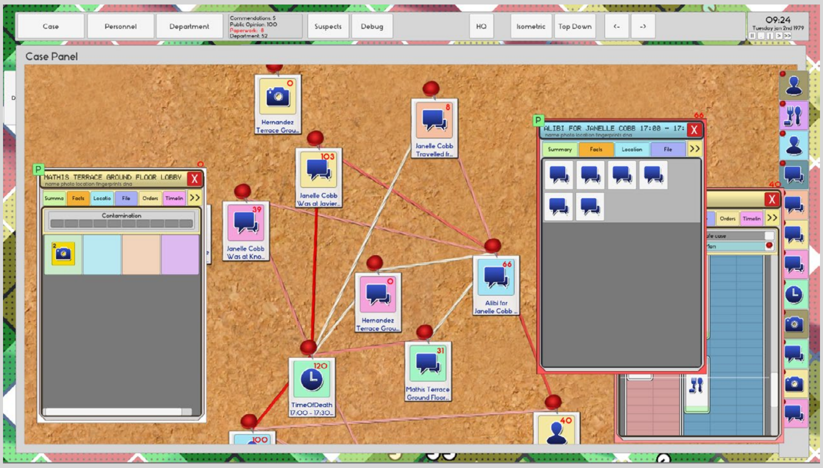
The initial appeal of procedural generation was to create a gameplay experience that could be different for everybody, and new every time.

I also like to design games that I like to play - and it's no fun if you know whodunnit…

The classic corkboard interface has made an appearance by this point; I spent a long time figuring this out, and at several points got sidetracked trying to turn this into a game itself.

Trouble is this is a potential huge rabbit hole in an already-ambitious project. Where do you stop? If you can do this, why not just let you go to the crime scenes and do everything yourself like a true first-person detective game? As awesome as that sounds, the line will have to be drawn in the sand somewhere. At the moment, the first person element is purely going to be an optional 'time waster' element that lets you feel more immersed in the game world.

It's going to be a challenge of course to create this world. At the moment the city is procedurally generated using a semi-but-not-overly complex system involving building density and zoning. For now I shall iterate on that generation, but later on, I'm leaning towards having the city as more of a designed element of the game. A city level editor at a later date would be a great inclusion, opening the game up to user-created cities, while keeping procedural cities as an option to the player.



Commendations: 5
Public Opinion: 100
Paperwork: -8
Department: 52

09:24
Tuesday jan 2nd 1979

Case
Personnel
Department
Suspects
Debug
HQ
Isometric
Top Down
<-
->
Case Panel
Hernandez Terrace Grou...
Janelle Cobb Travelled fr...
Janelle Cobb Was at Javier...
Janelle Cobb Was at Kno...
Hernandez Terrace Grou...
Alibi for Janelle Cobb ...
TimeOfDeath 17:00 - 17:30...
Mathis Terrace Ground Floor...
MATHIS TERRACE GROUND FLOOR LOBBY
name photo location fingerprints dna
Summa
Facts
Locatio
File
Orders
Timelin
>>
Contamination
ALIBI FOR JANELLE COBB 17:00 - 17:
name photo location fingerprints dna
Summary
Facts
Location
File
Orders
Timelin
X
P

I'm going to be pursuing that 'optional' design philosophy throughout the game. For example, all documents will also be procedurally generated and there for the player to read if they want to- but they don't have to. The useful bits of information are summed up in a handy list-style subsection, and represented by visual icons (think if somebody has summed something up via bullet points). It's really easy and digestible for the player- especially since the icons can link to another bit of evidence. Mouse over them and they'll tell you the fact; "Joe Burns lives at 3 Cedar Drive". Click on it and it'll bring up the file for Cedar Drive (or Joe Burns if you are clicking on it from the Cedar Drive file window).

But the extra detail is there for those that really want to get lost in the case- you can read the rental agreement between Joe and his shady landlord that the ‘fact’ came from.

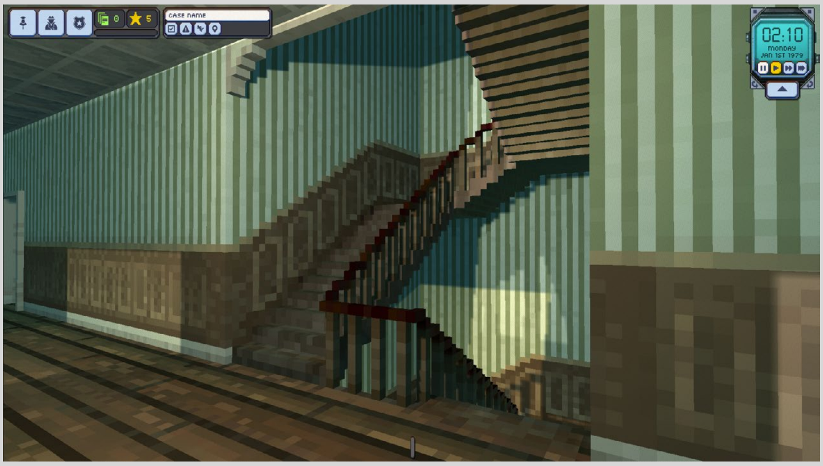
I love games like Deus Ex and The Elder Scrolls games where, if you want to, you can spend hours reading books- or more typically notes and other forms of interaction between NPCs. The love of those moments is one of the reasons I decided on this idea as my next project, and I plan to fill Shadows of Doubt with the same kind of detail.

If I had to pinpoint one of these blogs that marks the turning point between the old idea and what Shadows eventually became, it’s this one. The 3D world takes shape, and now, with a first-person character controller, the player can move around the world. These decisions informed large chunks of the rest of the design process very quickly; it was finally becoming clear what this project should be.

July 30th, 2018

This week has been hugely productive- following on from last week, I’ve implemented basic interiors so you can now go inside every building (complete with stairwells). That’s not to say interiors are anything like complete- there’s no furnishings, doors, or really anything of note, just rooms decorated in a sickly green striped wallpaper. However, the rooms even existing in 3D space is a nice milestone.

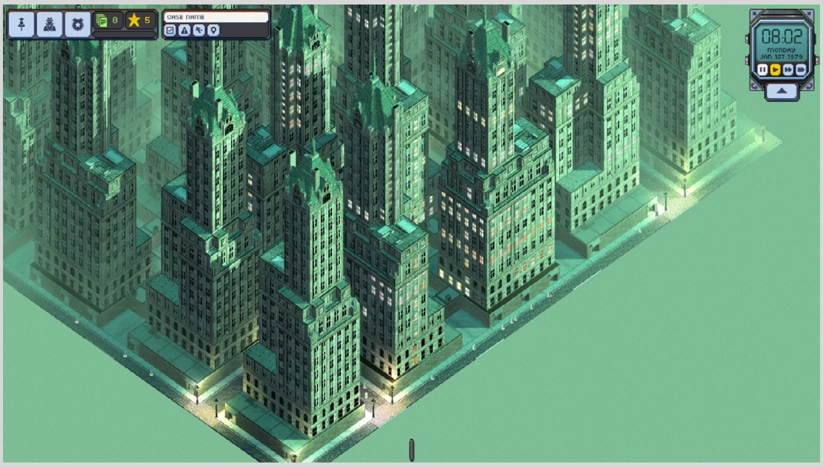
I’ve had to start thinking about the culling techniques for the game, as the game can’t possibly render every single room in every single building across the entire city at once. I’ve been able to appropriate my pathfinding/grid system to help me here, as interior space configurations are massively simplified by being on an internal 5×5 grid system, it’s able to help the engine know what to render and what to skip. The streets are rendered from inside a building, so you can see out the windows, but not vice versa. The only exception to this rule is the ground floor, which is rendered at street level, so it’s possible to have things like shop storefront windows which you can see into and out of. I’m happy with this compromise as it’s generally pretty tough to see into a window above the ground floor if you’re in the street.



I love the effect of the voxels creating such harsh shadows in the stairwells. It fits the noir theme perfectly!

The choice for the voxel art style started to solidify as the right choice here. The benefit of using voxels include the efficient development time and now, paired with a sucinct colour pallette it began forming the gritty noir style that acts as a foundation to the games art direction. The art direction that ultimately has helped the game become recognisable in a crowded indie market.

One decision I'm stuck on is what to do with the old default view; the isometric camera. As you can see below, the game looks great from this view, but sadly I cannot find much of a practical use for it. Currently, it's been taken out- the first person camera being essential for exploring, while the top-down camera is better for navigation and a general overview of the city. You can switch between them seamlessly with the F key. What the top-down camera can't do, however, is differentiate between different floors in a building. This could be where the isometric camera is useful.



The isometric view: Pretty but not practical for anything right now!

Lastly, with the focus now back on gameplay, I've been working on a first-person object interaction system. I've taken some inspiration from the original Deus Ex and System Shock here, as I'm sure anyone familiar with them will no doubt recognize. Maybe it's my own nostalgia, but in my opinion, it remains the clearest graphical way to interact with the game world. Not having a first-person weapon to design around frees up both mouse buttons to become available for world interaction- I've not quite figured out the rules for this yet, but it gives me some freedom in that respect.

This piece feels like the actual announcement post for the game. From here on out, things start to look like the game of today pretty quickly.

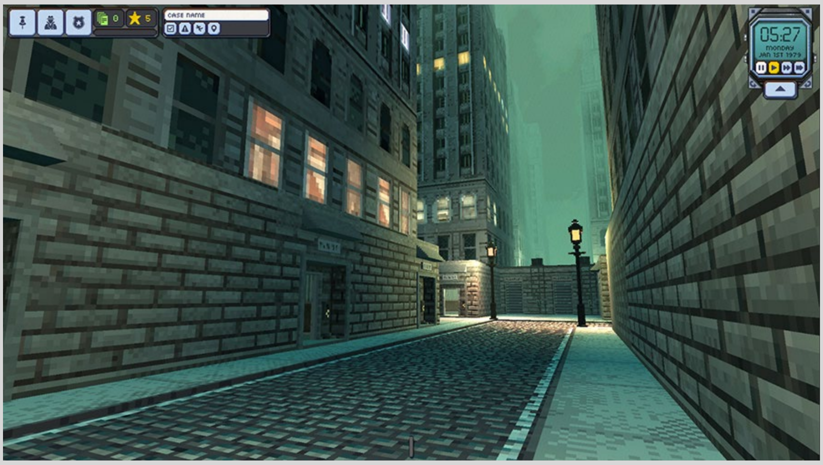
July 2Ist, 20I8

It's always been my personal game dev philosophy that I'd rather make games that are too ambitious for their own good, rather than under-ambitious. This project is not an exception. Despite my enthusiasm, however, so far the 'game' part has always seemed off. I love a good management game, but I guess a big reason why detective games and management games haven't crossed paths before (to my knowledge) is that… Well, it's pretty fun for the player to do the detective work, not leave it to a team of minions!

Moving away from the management game idea wasn't a decision I took lightly. At this point I was almost two years into the project. However, as detailed in the blog, I knew the game hadn't reached its potential. I knew there was opportunity for better gameplay. I highlight this because it's one of the harder parts of the game development process: Knowing when to move on from an idea, even when there has been a lot of time and effort spent. It did help that the game play became much more fun, quite quickly.

In the last update, I mentioned about putting a first person controller inside the 3D city that I've constructed, just to get a sense of what it would be like to explore the city from street level. Although my original idea for the game certainly didn't require that, part of what makes development fun for me is trying out those ideas. It's been a huge amount of work to create this simulated city and population, and I thought it a shame that you can't really feel 'among' it with a top-down camera perspective. I only really started with the overhead perspective because it's typically part of the management game genre. Often the reason for that is because the player is required to build things- that's not the case here, so it seemed ripe for experimentation

The hangovers from the old idea still loom large. Perhaps I was too cautious to change too much right away. As the development progressed it was clear a lot of the old ideas didn't fit. The game started to become more about the problem solving process of being a detective, rather than the technical aspects of the crime.



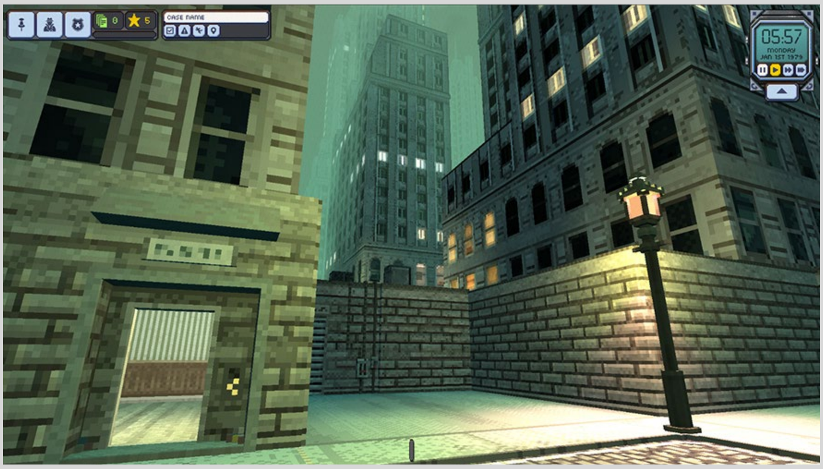
Shadows of Doubt: A first-person detective sim!

This is about the latest in the project I can comfortably change something as fundamental as this without having to remake a lot of things. The first person perspective is more of an addition than a change. I still want to keep the framework that I've made involving the procedural cases, population etc and even most of the management aspects. You can still pause the game at any time and interact with all the case files, as I've shown previously. The key difference being the player is now the detective. You'll still need your team of professionals to complement you- (e.g. forensics experts, ordering DNA tests, autopsies etc). But you'll be visiting the crime scenes, questioning and pursuing suspects yourself and generally being more a part of the world in addition to having that slightly lighter management role.

I'm really pleased these kind of details made it through to the final game. They convey a lot of important information to the player without being intrusive. They also add to the atmosphere of the cities.

Those building lights aren't at all random; it actually means the light has been turned on in that room (meaning likely someone is home).

This, of course, is a lot more work on my plate. As a bit of a visuals enthusiast though, I'm having a lot of fun making some of the early assets. I've chosen to go for a hybrid of pixel art and voxels to make the creation process relatively easy. I'm loving working with voxels- it's like working with Lego. I have a huge catalogue of building models from Concrete Jungle which I can adapt and 'voxelize'- it's saving me a decent chunk of time and because the art style is so different it doesn't feel like a cheat.



I'm pleased with the first building model in the game. Doors aren't a thing yet though!

Optimisation has been an ongoing challenge throughout the entire project. Especially with the console ports. Luckily we have managed to work through a lot of the technical limitations of having completely dynamic levels. I did not want to compromise the scale of the world, limiting the light count has been a way to achieve the scale I was aiming for.

Because of my 'everything-is-simulated' approach, the procedural generation means shadow baking is unavailable to me. The lighting is going to be a bit primitive as a result, but luckily this works fine with my low-fidelity art style. It also means all the interiors have to exist in some state- when I've got skyscrapers with 20+ floors, this is a challenge.

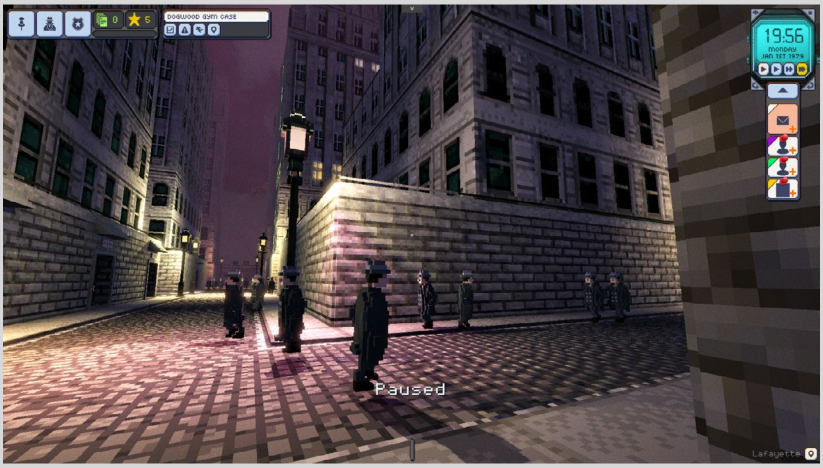
August 18th, 2018

Last update I had made some progress on the first-person game world and these last few weeks have continued that trend. I'm enjoying working with voxels so much that switching to 3D modelling for the game actually feels like a nice break from the programming, so it's a win-win

Things are really taking shape now, and I remember this phase of development being full of joyous experimentation within the world. The city itself didn't take long to look decent; so attention turned to the characters (which we'll encounter again later).

Later in the development process Miles was able to identify and implement important visual cues that give the characters personality and interest. Before that, they began as identical clones walking around the city, going to work and going home and starting it all over again. Having a city full of stiff clones was actually the creepiest form of the city, and murders hadn't even happened yet.

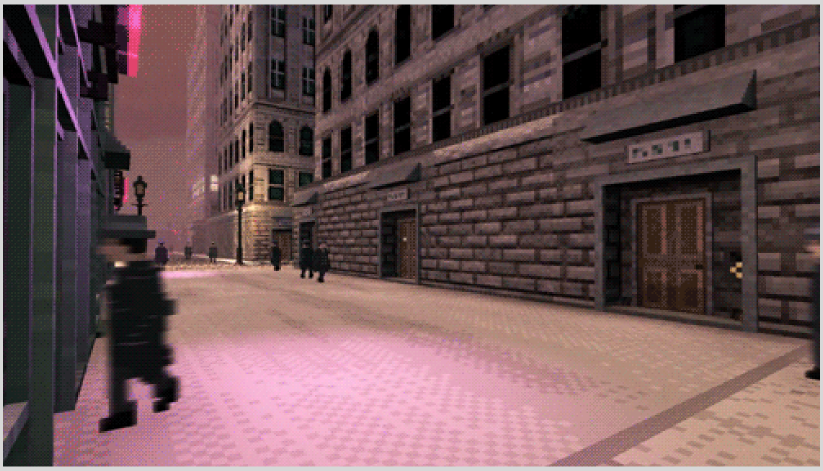
situation as far as development is concerned! It's why the graphics suggest the game is further ahead than it is. Since the last update, the biggest implementation is a basic character model, complete with walking animation.



Despite the endless repetition, having a human model in the game has really brought the first-person mode to life.

Originally, I had planned to make the characters as 2D billboard sprites in a 3D world (think original Doom). However, after some experimentation, I decided to try and make a character model using voxels and the same technique I'm using for, well, everything else. Turns out 3D voxel modes are actually way easier- for several reasons: The first being that it's much less time consuming to animate compared to drawing every single frame for every animation for at least 4 different directions in 2D. Secondly, although obviously right now the city is pretty non-diverse with my basic film noir white detective, I have big plans for having all shapes, skin tones, hair colours, and hairstyles etc. Building the character models in a way that is easily segmented into head, torso, and legs means this diversity among characters will be possible, and that's super easy to do with voxels.

The walking animation was achieved by making different leg meshes for each' frame' of animation. This achieves a stop-motion-like effect. Since the legs are symmetrical, I was able to save time and resources by simply mirroring the mesh in the latter half of the walk cycle to complete the animation. I then added some subtle transform movement of the torso and head meshes using Unity's built-in animation editor. It's by no means final, but honestly, it's better than I was hoping floor (I don't have a lot of experience in 3D animation at all, and I was dreading the task making all those billboard sprites).



You'll hear much more about the characters later, but at this point it was just about getting some kind of visual representation of them within the game world. Many hilarious bugs ensued throughout development; from having them spontaenously change clothes when attacked, the killer immediately panicing and reporting the body as soon as they killed. Or citizens having casual phone conversations about last nights sports game just after discovering the murdered body of their spouse.

These last few days saw the first implementation of doors.

But wait there's more! This week I've also made some doors for the game. Doors in Shadows of Doubt will be able to be closed, open, locked, unlocked, lockpicked, destroyed and keyholes peeked-through. I've only got the first couple of things on that list working so far, but the implementation was relatively painless, and it all seems to be working as planned for now.

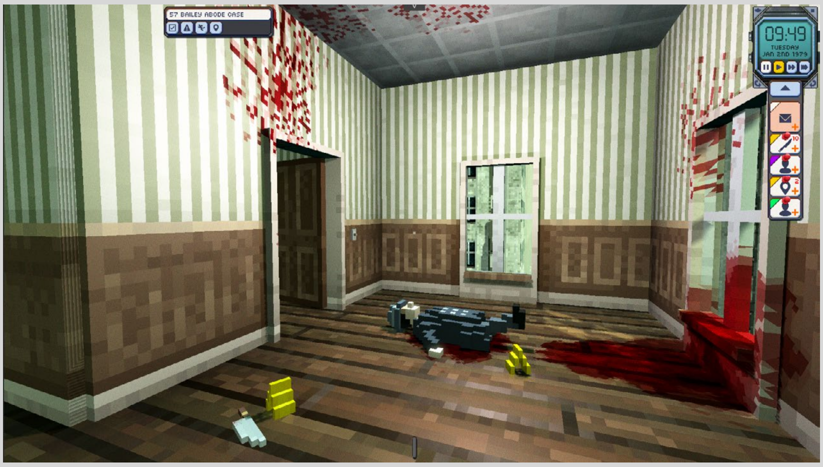
I mentioned in my last post, one of the limitations of having a procedural world is not being able to bake lighting. I've already come up with some culling techniques for expensive real-time shadows and illumination. One nice advantage is though, I can have dynamic lights being visible through cracks under doors- a very noir-esque way of telling if the light is on in the room within. This actually happened by accident first of all due to wonky door positioning on my part, but I'm absolutely keeping it!

As a developer, I have learned to embrace the occaisonal gift of having something unintentional work out for the better (this is one such incident). There haven't been a huge number of cases of this happening in development; but it is satisfying when it happens.

February 28th, 20I9

For the last few weeks I've been working on an important part of the game; the environments. One of the most ambitious parts of this project is the huge number of explorable interiors and their complexity. I've had to think very carefully on how to go about creating this:

I need a way of generating huge numbers of apartments, offices, cafés etc that have enough variation but also don't require a huge amount of individual design work. It's perhaps the toughest aspect of the project, but I've had a long time to think about it and progress is going well.



Interactivity has always been important to me in terms of world building. It might create more technical limitations but it's way more fun than having a static environment! A lot of the objects we added to the world, even at this early stage, were interactable. This set a high bar, but along the way created some natural gameplay hilarity: Adding drinkable acohol for example allowed players to role play as the ‘drunk detective’ trope.

What makes a location in a game interesting? With a narratively-focused game, I find that environmental storytelling can really make a location feel compelling. Empty beer bottles. Broken mirrors. Books lying around that you can read snippets of. Pictures on the wall. Notes on the fridge. The environment is not only part of the player's world, but someone else's. With my citizen generation system, I have a lot of what I need in place to drive environmental storytelling. Time will tell how compelling that can really be, and I'll be writing a lot about this in the future. But for the moment I've been focusing on the interior blueprints themselves.

Buildings in Shadows of Doubt are split into floors, then divided into ‘addresses’, and within them ‘rooms’. The location hierarchy in the game from large to small looks like this:

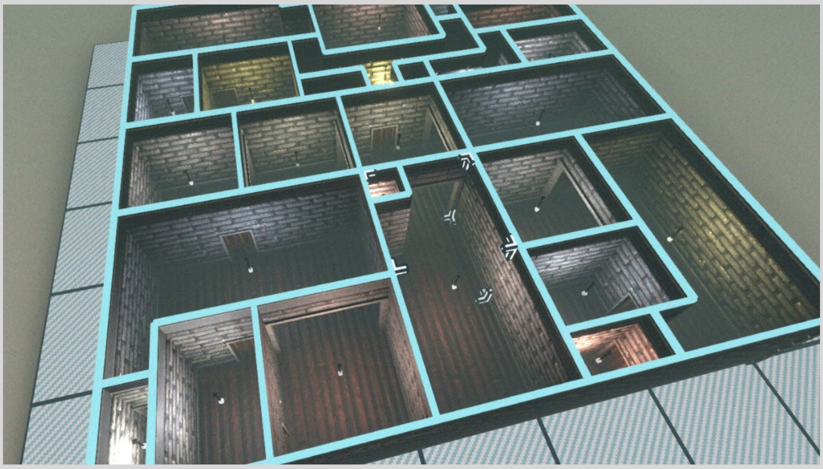
City › District › Block › Building › Floor › Address › Room › Tile

I decided to concentrate on ‘floors’ as the point where designed elements intersect with the procedural generation. I set to work on a separate tool that lets me design floorplans, while also letting me preview what the procedural generation can do. The idea is that I design lots of individual floorplans that are sectioned off into different addresses- kind of like in SimCity where you lay out residential, commercial and industrial zones. Instead of that though I'm defining the floor space, of say an apartment, within the floor of a building. Buildings can then be generated from a bunch of these floorplans.

This is as far as my input goes and where proc gen takes over: I have a 'generate interior' button that uses some wizardry to generate dividing walls. In this example with an apartment, it will section off a bathroom, and if there's room, bedrooms, a kitchen and for much larger floorspaces even a study, shower room, dining room etc

How does this work then? The physical space in the game is represented through 'tiles'- each one of these is I.8m x I.8m in-game, basically turning everything into a giant grid that makes it easier for many systems such as pathfinding, culling and this generation process. Building floors are I5 x I5 tiles in size.

When generating an interior layout, my system takes stock of the floor space and where the entrance to is. It decides if the entrance is too far from a corner of the defined space, to draw a hallway that connects the entrance to a tile closer to that corner. Hallways are essential because without them you tend to generate a bunch of rooms that are only ever accessed via each other. This might be fine for something like a small apartment, but as they get larger the less sense they make. Imagine having to get to the bathroom by having to go through the living room, then the kitchen, then the study, then a bedroom etc.



Not a huge amount changed between these posts and the game's final form. Regrettably though, all my hard work making the procedural rooms system talked about here did turn out to be a little bit redundant. It turns out that applying the process to an entire city of rooms was extremely intensive and made generation times intolerable. Perhaps even more importantly this proc gen technique only spat out a few viable combinations of rooms anyway. Lots of heavy CPU work for something that can easily be pre-computed. It's a lesson in why you usually find this kind of proc gen used for caves, dungeons or other non-contained areas. It made much more sense to generate the few different room combinations and embed them in floor data itself, then simply pick between those combinations when the city was generated. It's a bit less cool but just one of those practical decisions you have to make in game development. The proc gen furniture placements and decoration colourings made it through pretty much as described and stands out as a personal highlight for me; I love exploring how the game has decorated and furnished apartments.

Next, the game will cycle through a list of defined rooms, each with their own pre-defined rules. It cycles in order of how 'important' they are, e.g. the first entry for an apartment is a living room, followed by the bathroom (I think they're pretty much the most important rooms!) The game will then loop through every available free space and simulate positioning the room there. It will rank how successful or unsuccessful this has been using the rules. It will then save this placement location and ranking in a list. After it has tried every available space, it will sort the list using the ranking and pick the most successful attempt, and place the room there. Repeat until either all rooms are placed or there is no space left. I currently have 3 main means of ranking how appropriate a room placement is:

I). Floor space. Important for larger rooms, for example the living room. Less important for say a bathroom.

2). Uniform shape. Basically how many corners they have. This is important for most larger rooms, maybe less so for smaller ones.

3). Windows. It's more important for some rooms to feature windows than others. For example it's more likely your living room or bedroom will feature windows compared to a study.

That's the basic model, but I've had to add a lot of special conditions to this to make things work appropriately. Certain rooms can only connect to other certain rooms. For example, while everything can connect to a hallway, a kitchen can also connect to a living room directly. Some rooms can have only I door (e.g. bathrooms), while others can have more. Some rooms are allowed to feature the entrance to the address (you probably don't want the main door to open up into the bedroom). There is also an override system that allows rooms to 'steal' floorspace from existing rooms in the generation process if for example one previously placed room has excess floor space and the current one has too little. After a lot of tweaking, I think I've found a decent balance. Of course, you can also preview what's been generated in the editor too!

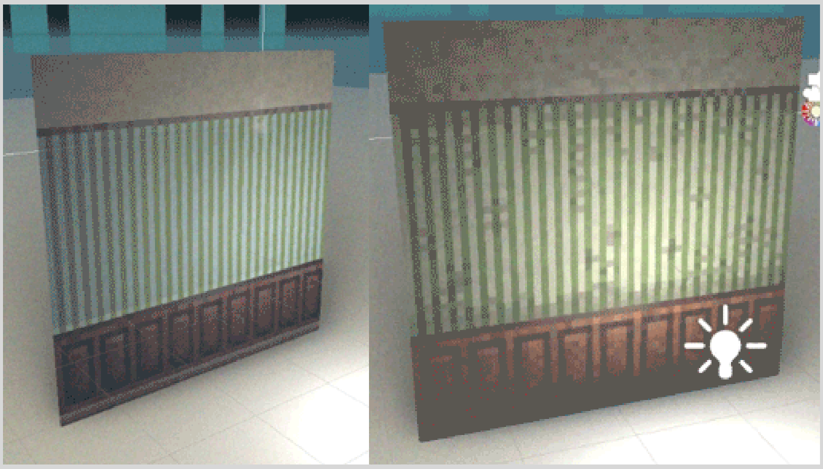
March I9th, 20I9

As you may have seen in previous entries, I want the game to have a mix of interiors ranging from old and murky I930s decor to more modern 70s and 80s styles (the game itself is set in the 80s - albeit an alternate reality).

The first task was creating a custom shader that could take walls, floors, even props and apply different colours to them. Manually creating unity materials for each of these, then a range of colours wouldn't really be feasible due to the sheer range of them I would need. Creating a custom shader with a map that told where to apply certain colours seemed to be the way to go. I'm a novice when it comes to writing shaders, but thankfully Unity's new shader graph came to the rescue here, and it didn't take me too long to come up with something that functioned as intended. Remember that green striped wallpaper that was applied to every single interior wall before? Well, now I can have that in any colour combination I want. Progress!

The addition of the dirt map made the world feel more lived-in. This was an important decision in creating the atmosphere of the game and leaning towards the noir style.

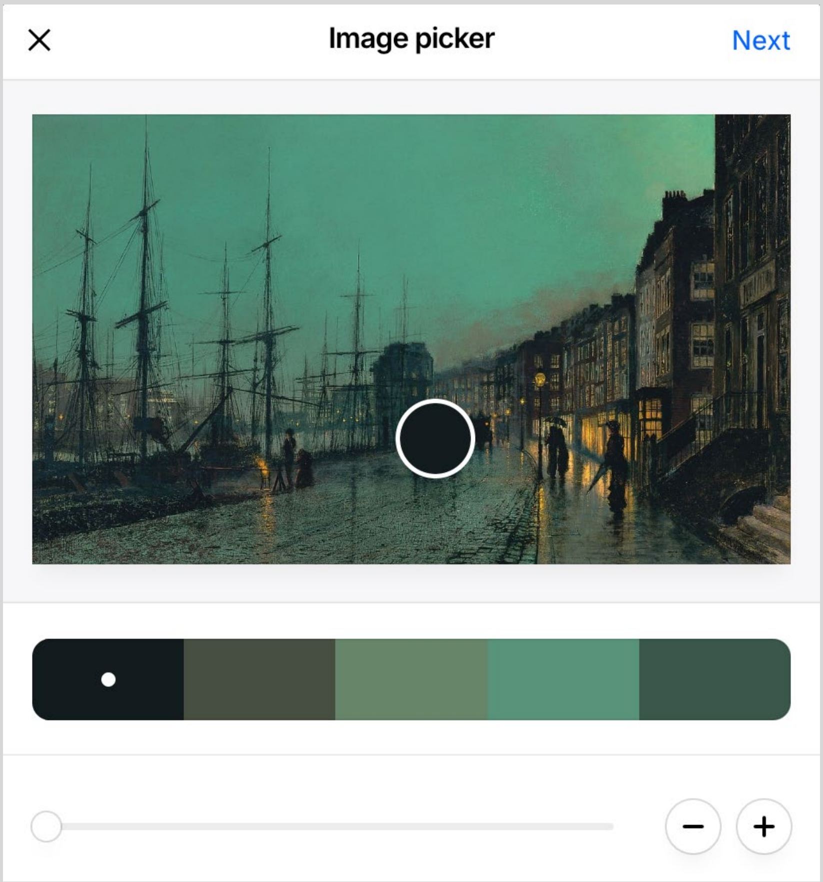
In addition to this, I also wanted to represent neglected/abandoned apartments by having an additional dirt map that I could apply also using the shader…



Now I've got a base material that I can alter using a couple of simple inputs. But how do I get my game to look like the style I want? I would have to come up with a system that selected colours similar to the designs I mentioned a couple of months back.

I decided to make a custom palette that features 5 colours: 2x Primary and secondary colours that compliment each other, along with a neutral colour. I used scriptable objects, so I could easily create palettes in-editor.

After some messing around I found a great way to pick these 5 colours. Using images found on various sites and design books from the eras I want to mimic in game, I use coolors.co to pick a complimentary palette from them. It's a great free tool that allows me to cycle through several palettes from each image to get one that I like. I can't export into my scriptable object directly, but copy and pasting the colour hex codes doesn't take long!



As I also wanted the game to pick from these using citizen's personality data- as I wanted someone's apartment to reflect their personality to some extent. So I also added some variables to the palette object that enable me to do this. I settled on these 4 variables:

Modernity: How 'modern' this colour palette is in terms of design. Muted colours like the original green striped wallpaper may be suited to a less modern style, while brighter reds and oranges might be more suited to more

Using the personality data is what helps the procedural genereation feel more natural and designed rather than randomised.

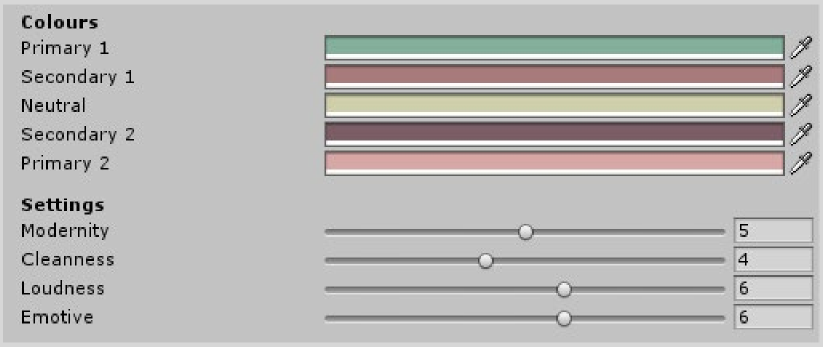
It was important to create rules that made the procedual generation a feature of the game play and not the focus. There are definelty limits and constraints when creating a world that creates itself. But I think implementing rules and categories like this have helped to shape the final product.

modern styles. Richer citizens tend to be able to afford more modern decor styles.

Cleanness: This one is more to do with the room type that citizen personality. ‘Clean’ palettes will be found more often in bathrooms or corporate environments.

Loudness: I couldn’t think of a better name for this, but in broad strokes, this is how bold the colour scheme is. Bright reds, for example, are ‘loud‘, and are more likely to be found in apartments of more extroverted citizens.

Emotive: In general how ‘warm’ or ‘cold’ the colours are. This is similar to the cleanness value but links to personality instead; more emotive citizens are more likely to feature these palettes.



Colours
Primary 1 🟩
Secondary 1 🟥
Neutral 🟨
Secondary 2 🟪
Primary 2 🟥

Settings
Modernity 5
Cleanness 4
Loudness 6
Emotive 6

Now when a citizen is assigned an apartment, I can use a system that matches colour palettes to their personality!

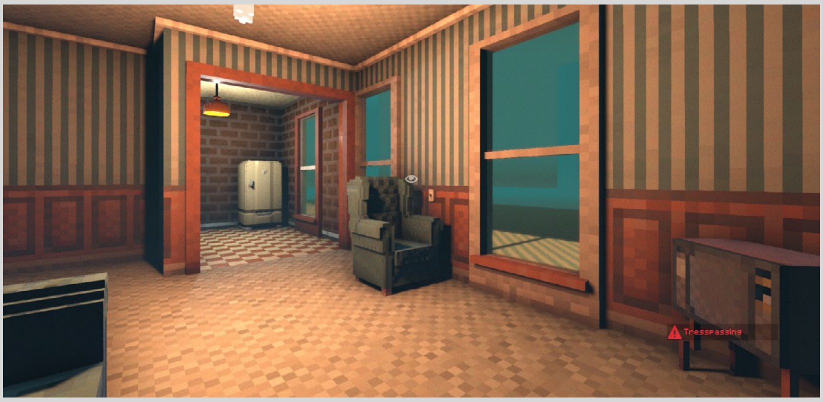
I’ve developed a very similar system that does the same thing but for the base materials themselves: A floral pattern may be considered more emotive for instance. A combination of these methods to procedurally decorate apartments, offices etc is a great starting point that gives meaning, reason, and personality to the procedural generation that is at risk of becoming too random and cookie-cutter. Hopefully, this kind of attention to detail will excite some of you (and I bet a few think it’s pointless!).

It’s already starting to make my previously labyrinth-like environments look like something more habitable. The next step is to start work on some basic props: Chairs, tables, beds etc.

In the very initial stages of designing basic props I used inspiration from my environment. A vintage sideboard from my living room and pictures of my dog still exist in the final game.

April 24th, 2019

The previous dev blog covered my first foray into interiors and my first attempts to procedurally decorate them. I'm pleased to say the game is progressing at a decent rate now; whereas before the new interiors were limited to my floor editor, they are now actually in-game. It took a while as there's a lot to integrate; one of SoD's main selling points is complete freedom to explore- and that means being able to go into any room in any building. When you're talking about even just a couple of city blocks worth of skyscrapers, that's a hell of a lot of rooms! I've had to do a lot of work under the hood in terms of culling to get this working, but you can, at last, do this without any extra loading times. I've got a lot to refine, but it's working well. Using low-fidelity assets really helps as it keeps texture sizes small.

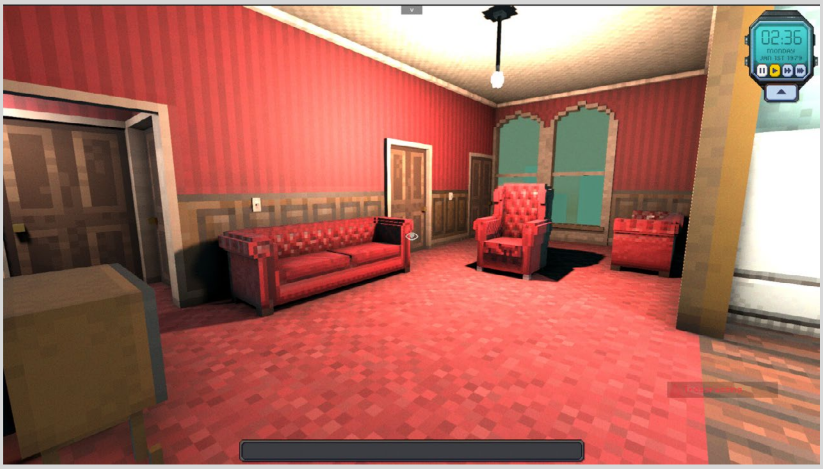


The first batch of furnishings are appearing in apartments, giving them some much-needed character.

At this stage I was still grappling with what the AI had to be to function in a game like this. It's probably the most complex portion of code in the entire project.

The second hurdle was to re-write the AI to take advantage of the new interiors system, as they've completely changed to how they were prior. I opted for an overhaul of the previous very basic AI in favour of a new goal-based system. A lot of things have stayed the same, such as the way the AI sight each other, but they now employ a new sims-like stat system that dictates their minor actions throughout the day (needing to eat, drink, sleep etc). Whereas before their routine was almost completely pre-computed- meaning a lot of extra, spontaneous actions such as checking out a suspicious noise or sighting had to be written separately. This could potentially break their pre-computed routine, meaning the game would have to calculate it again. This new system allows for a lot more flexibility.

As the goal is still to have a lot of citizens simulated with this AI, I've done a few things to limit the performance cost of it: Most notably, the AI's update rate will only be high when nearest the player and when not doing actions that require them to be more aware. By dynamically changing this with proximity to what's actually on-screen, I can make the performance cost of 95% of the citizens at any one time relatively insignificant. It's had some teething problems, but I'm almost there- and hope to complete the new AI around the beginning of May.



Most significantly for the game though, this means I'll finally have furnished, individual rooms that you can sneak into, complete with simulated citizens that are capable of interacting fairly realistically. This is a pretty big milestone that I'm super excited about. I'm hoping to launch the Steam page around then as I feel for the first time I'll be able to effectively communicate (show, not tell) what the gameplay is like. Up until now, my descriptions have been somewhat vague, as I've tested a lot of ideas throughout this development. But what I'll have soon is a stealth game with some pretty clear mechanics, that although lacking a lot, will be the first thing that's actually somewhat playable. The sand in the sandbox if you will. Only took me 2 years!

September 6th, 2019

I've been working on quite an exciting new feature for Shadows this month: Procedurally generated ventilation shafts! If you don't play many games then that

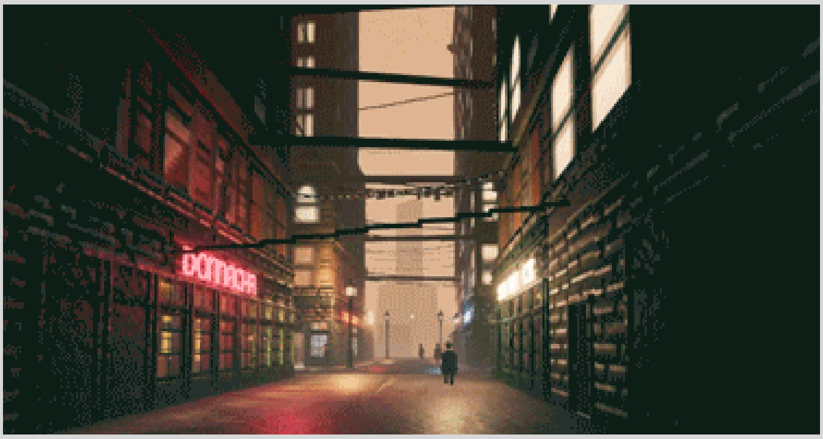
Air ducts and ventilation shafts became an important feature for game play: Although by no means realistic, they add an important covert exploration option to the players arsenal.

Stealth creates peril for the player; something that felt lacking in very early versions of the game. By making somewhere where the player wasn't supposed to be, it ramps up the excitement levels.

Therefore, creating oppotunities to intrect with buildings and objects in a stealthy way was an important part of the environmental design process. The vents lend themselves well to the industrialised, distopian setting of the Shadows of Doubt world. They have been a tricky part of the procedural generation, leading to some interesting bugs and floating detectives but they pay off has been worth it. They help link locations that would otherwise be difficult to get to and they are an excellent opptunity to add some great sound design too.

might sound incredibly boring, but to any seasoned FPS or immersive sim player then hopefully that sounds excellent.

One of the biggest things missing from the gameplay right now is multiple ways to get in and out of locations: The choice-driven stealth gameplay I'm shooting for requires that the player is given a choice in how to infiltrate their places of interest. Before this update, there was only the front door, and in the future, there will be yet more options including fire escapes and windows. But one of the biggest tasks on my to-do list was these air ducts that the player can crawl through to get places they shouldn't.

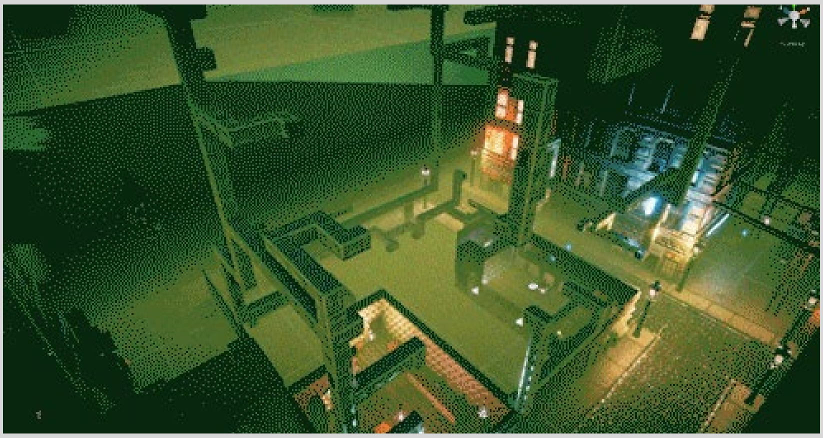


Originally the ducts appeared on the exterior of buildings by accident, but I figured they looked way too cool to get rid of.

Previously in the project, I briefly toyed with having a more realistic air duct system as described in this article. I have kept a few issues in mind, although in the end, I decided to prioritize gameplay over realism- and with that I decided to leave my air ducts at the mercy of procedural generation. The air ducts are completely procedurally generated within each building. Even if two or more share a similar floor plan then the shaft system within each will still be completely different, and your infiltration routes and options will be different too.

Going into the technical side of things a little: Basically, in the configuration settings I have for each room, I've added range variables for how many air vents can be added. For example, a bathroom might have one, or a kitchen, but less likely a lounge area. So when the rooms are generated they will also look for a place to put an air vent; often in the ceiling but sometimes high up on a wall or (less likely) closer to ground level on a wall. So now we've got all the points where we want the vents to be in the building.

Next up I map all the available space that the ducts can use. Mostly this is the space between ceiling and floors, but it can also cover spaces inside maintenance closets and even space outside the building. Now I have this space mapped out and the points I need to connect, I can simply use an A* pathfinding routine to connect points together- this becomes the path of the ducts. With every path calculation, I can weight already-placed ducts so the game won't create air ducts which run immediately parallel to each other.



This is what the duct system looks like minus room and building models.

As with the walls/floors/ceilings, I have a different model for each possible turn/junction of duct- this is how it's possible to create them. They all share the same material within unity, so I can easily combine them together to form one model and keep performance high.

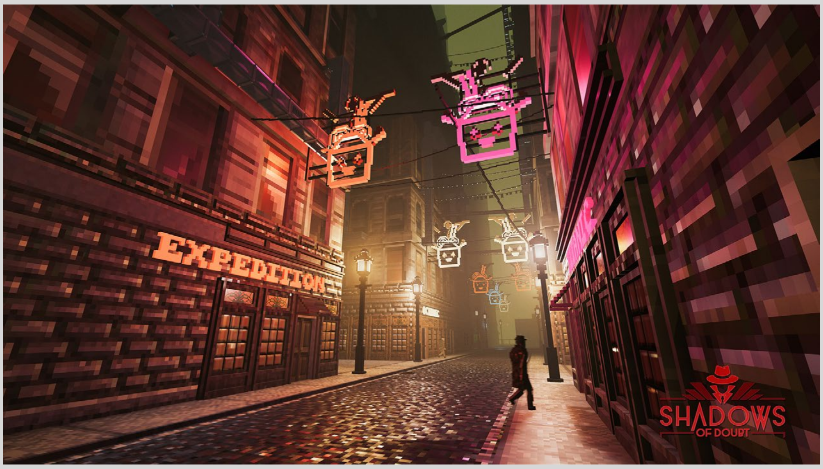
It's working really nicely! When inside the player can freely climb up and down vertical duct sections too. One last thing that I need to work on (aside from a couple of misc wall bugs) is enabling ducts to be discovered and mapped on the minimap. It would be cool to maybe be able to discover blueprints that map out the ducts and vents of the whole building- adding incentive to research your places of interest before infiltration.

Anyways, that's it for this update. Progress is going nicely on the game- next up is a focus on improving the AI some more and fleshing out the side mission I showed off a few weeks back. Stay tuned for more soon!

March 27th, 2020

From the confines of coronavirus lockdown, I bring you a brand new and over-due Shadows of Doubt development diary!

There have been points over the last 2.5 years in this project where I've felt like giving up; it's such a large ambitious project that at times it has felt like I had bitten off more than I can chew. I'll likely write about the challenges in more detail with another post, but for this one, I wanted to share with you perhaps the biggest saviour in terms of production viability: Voxels.



Back in the pre-art asset days of this project, when it was still a management game, I often wondered about what direction the art style would take. Realism was off the table due to workload, but I really wanted to explore a pixel art approach. As the game shifted to 3D, and then entirely to first-person, voxels started looking like the way to go.

This turned out to be perhaps the biggest decision in actually making this ambitious project actually somewhat do-able with a small team and small budget: The reason being that the turnaround of most art assets is minuscule in comparison to anything else.

Another benefit of voxels has been the ability to make fairly horrific amount of blood splatter more palettable!

Deciding voxel resolutions was an odd problem: A purist would argue for one resolution throughout the entire game. Given the scale of the contents of the game world however, it quickly became clear that was impractical: If you have a small item like a bullet casing and then apply even a modestly low resolution to that but then make an entire building in that resolution you are dealing with an incredibly detailed and high-polygon model. So for practical reasons we had to decide on several different voxel resolutions that were applied depending on the scale of the model itself.

It's arguable that the voxels are a little bit of a mismatch in terms of what people expect. They're associated with Minecraft and a general cartoonishness, which doesn't fit the tone of this project at all. On the other hand, they do effectively evoke a low-fidelity style, something which has recently taken off in some really, really cool projects that I adore. I think in an ideal world I would choose a low-fi, low ploy art direction over the voxels as it's more effective at conveying the atmosphere that I want. But crucially I also believe this would have resulted in increased turnaround time in regards to art assets. I'm happy with the trade-off.

When used in conjunction with the unity high definition pipeline it really pops. There's something about the use of voxels and modern render technology that makes something look really cool. I'm not sure how else to describe it, because logically the two should be at odds with each other? Maybe it evokes the way we remember old games of our past; always looking better than they actually did. As if they were brought to life, but not replacing that low definition that allowed us to fill the gaps with our imagination.

CTION
SHADOWS
OF DOUBT

Voxels, then. After a bit of research, it became clear that the main contender of voxel software is MagicaVoxel, an extremely awesome bit of free software that pretty much all voxel artists use. Great, that's an easy decision then? Well no. Although it does a whole bunch of stuff really well unless they've changed it since I last looked at it, it doesn't do two very important things that I figured I needed early on in this project:

Be able to convert, or 'voxelize' traditional 3D meshes into voxels. This is important as I decided quite early on, to make this manageable I wanted to re-use some of the 3D building models I made for Concrete Jungle for this project.

Be able to output voxel meshes with traditional UV maps instead of an atlas. This was important as I wanted to make normal maps for my models, and not just have them all as a flat texture. This was important in order to move away from the cartoon flat visuals and towards the low-fi look.

Then office co-worker Nick Gunn, who works on Industries of Titan (which uses voxels to crazy-awesome effect) recommended looking at Qubicle. It can do both of these things and also has the added bonus of being quite good at optimizing meshes for use in Unity: Something which magicaVoxel at the time also lacked.

SHADOWS
OF DOUBT

There was a short learning phase; at first, Qubicle being limited to isometric view really bugged me, but I soon got used to it. I also began to establish my workflow. What was the best way to go from nothing to a final in-game model? My original vision involved using a pixel-art setup in Photoshop to manually edit the outputted Qubicle texture maps. I would use a really cool colour indexing technique to make everything look more like pixel art than anything else. It kind of worked in practice, but it soon became clear that to produce effective art assets quickly, I really needed to be able to paint directly onto the model. Photoshop does have this capability, but frustratingly there is no option for using point

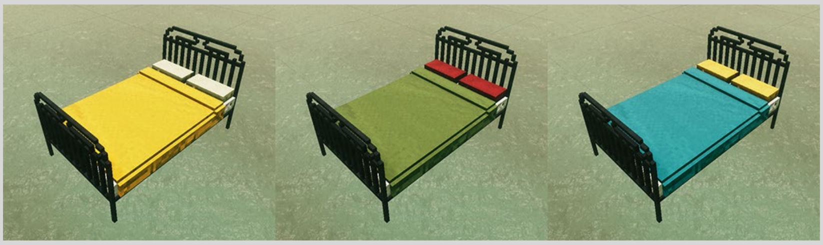
You'll hear much more about this later as Miles, the game's lead voxel artist adopted this workflow.

filtering, so my pixel art was lost to a horrible soup of texture when projected onto the model in Photoshop.

I explored some other options, but frustratingly I couldn't find anything that allowed me to UV paint and that didn't force the texture to be blurred. In the end, out of ease more than anything else, I decided to import my pixel art colour palettes into Qubicle and just use that to make the texture maps too.

Actually, after a brief adjustment period, I grew to really like it. It's pretty simplistic – nothing fancy. But it has the essentials, plus it's quick and easy to add noise, which is nice as it again makes it easy to avoid the flat surfaces that look cartoon-like. It was probably the tool I was looking for all along for both modelling and texturing. As I grew more used to using it, prop creation time reduced dramatically, and now most basic props can be created from scratch in under an hour, and for smaller things about half that.

Unfortunately, Qubicle doesn't let you output the texture map alone, so my workaround for creating multiple maps involves exporting my original model with a colour map texture, then duplicating it in Qubicle and turning it to greyscale and creating a heightmap with it. Then, I export this as a separate model (and along with it it's UV texture). I sometimes do this a third time to create a smoothness or metallic map too. Then it's a case of putting all these textures together into a unity material (unity can automatically create a normal map from a height map) and we're done!



The final bit of magic sauce I use is a special custom shader than can colour things without needing a whole new texture. I've talked about it in the past here, but what it does is allow another texture map to define unique colours to apply to the model. So for example, I can take a model of a bed and apply a special texture map to it that keys out the pillows in red and duvet in green, That 3D artist we added to the team was Miles Atkinson; a young pixel artist who joined the team over what turned out to be the initial wave of the covid pandemic. As voxels were and still are such a niche style, I decided to look for a pixel artist who would be keen to adapt their talents to 3D. Truthfully, I enjoyed creating models myself but the sheer number required for the project far outweighed what I could achieve alone. This turned out to be a fantastic decision as Miles bought with him an expert eye with attention to detail and churned out many fantastic models over the course of the next 18 months or so.

By this point in the project, we were now signed with a publisher so our deadlines got a little stricter and the day-to-day development slowly switched over from experimentation to refining and implementation (slightly less fun to write about). I published development blogs less frequently from now on as the next several years were spent butting heads with technical and duller design challenges. I spend less time on the visual side of things, as Miles takes over that role, so as a result you'll hear from him about his work on the game...

with the rest black. The shader will colour black areas with the base texture map, which will be the same for all bed model instances. The red and green keyed areas, however, will draw colours from my pre-defined colour palettes which consist of 5 colours and are generated for each individual room. This is how the game can generate interiors with colours that are complementary to each other. With enough assets and this technique, I hope that we're able to move away from the cookie-cutter effect you see in many procedurally generated games.

At the moment you've probably seen the same few props hanging around in my screenshots, but over the course of development during the next year or so you should see this greatly expand. Not least because we're planning on adding a 3D artist to the team this year.

Block by Block

Voxel art process by Miles Atkinson

SHADOWS OF DOUBT
DEVELOPER PROFILE

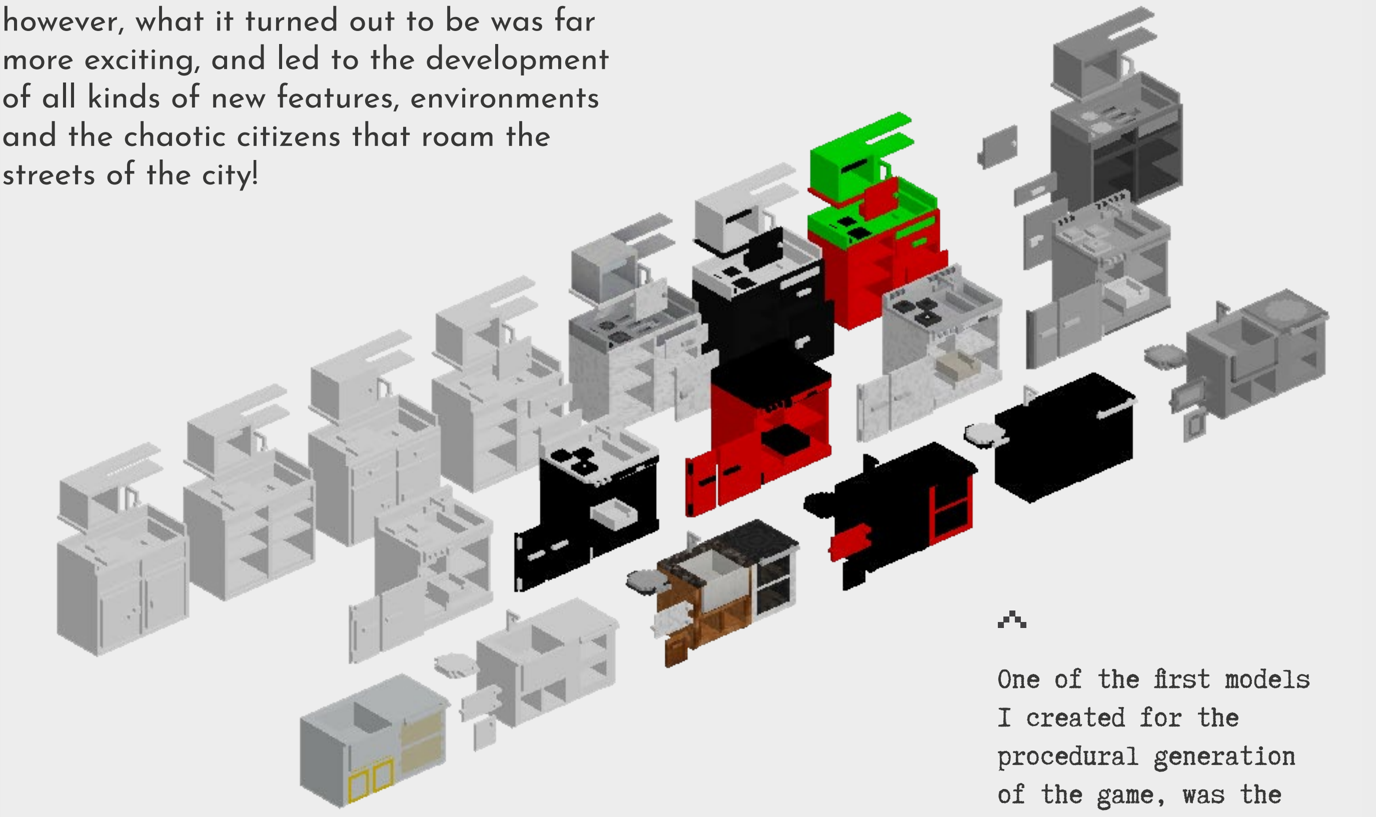
Miles Atkinson

Voxel Artist
Shadows of Doubt

In-game undercover name:
"Jude Hurd"

EARLY MODELLING

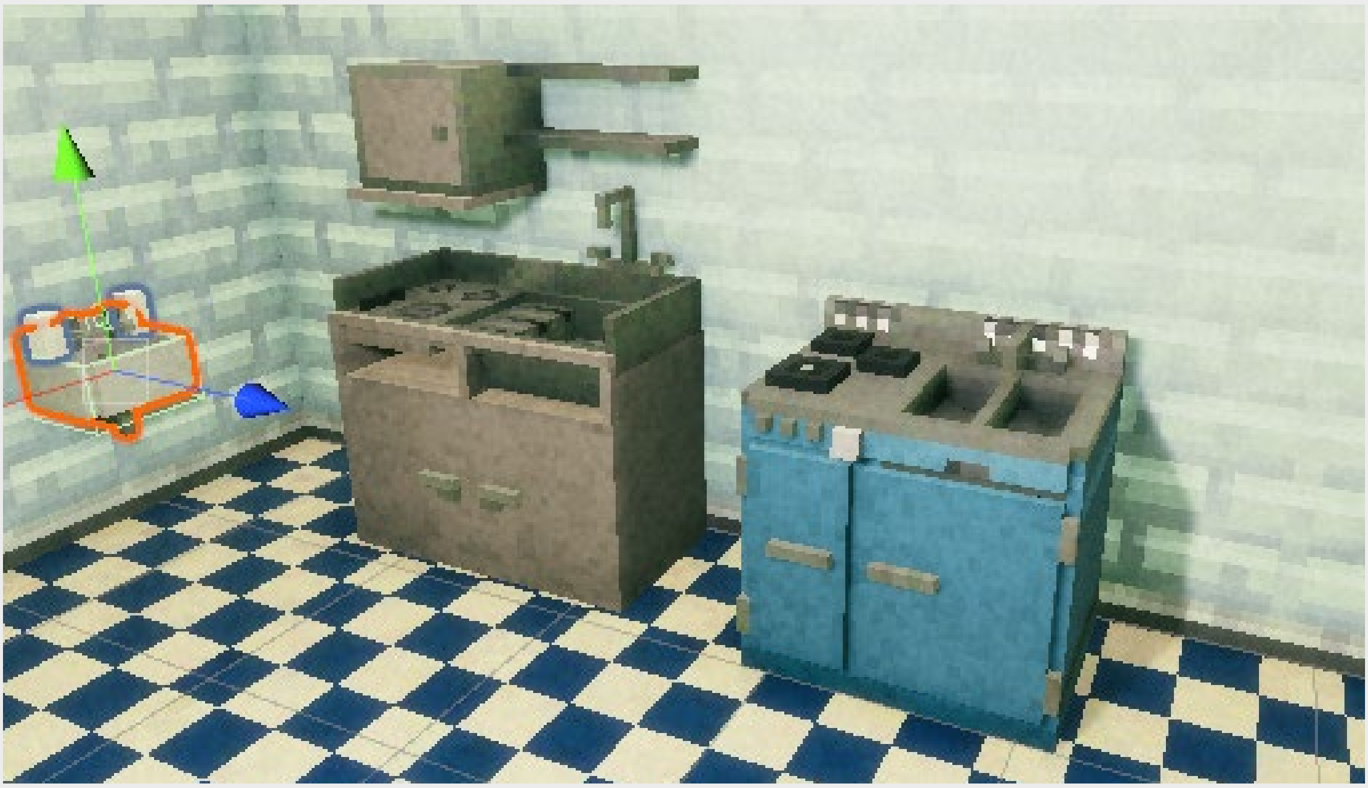
In October 2020 I joined the project to work as a voxel artist. My job was to create new props to help fill out the procedural generation of the world. At first I thought this would just be furniture and items, however, what it turned out to be was far more exciting, and led to the development of all kinds of new features, environments and the chaotic citizens that roam the streets of the city!



One of the first models I created for the procedural generation of the game, was the kitchenette. For which, I created 3 different styles - vintage/early century, I960s-70s, and modern.

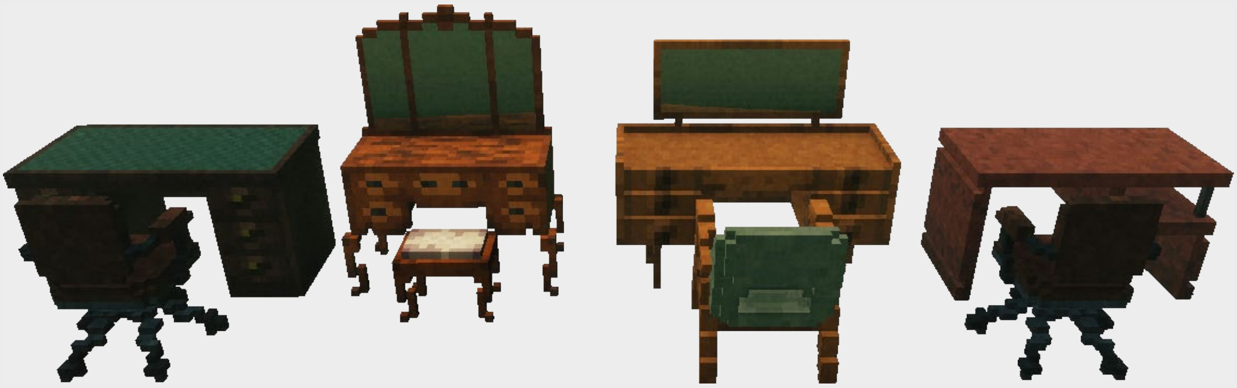
There's very little space within the apartments of the city, and therefore, an all-in-one style of kitchenette, offering both cooking, cooling and storage spaces is ideal. No adults, children or elderly are allowed in the inner city in which the gameplay takes place, most apartments are either singles or couples. Therefore, there's also not much need to have lots of storage space.

In the screenshot above you can see my process of modelling. It starts off with a simple white boxed design, which I then colour and texture, before duplicating and splitting off parts, such as doors, and creating diffuse maps (a greyscale version of the initial coloured design), colour maps, height and smoothness maps.

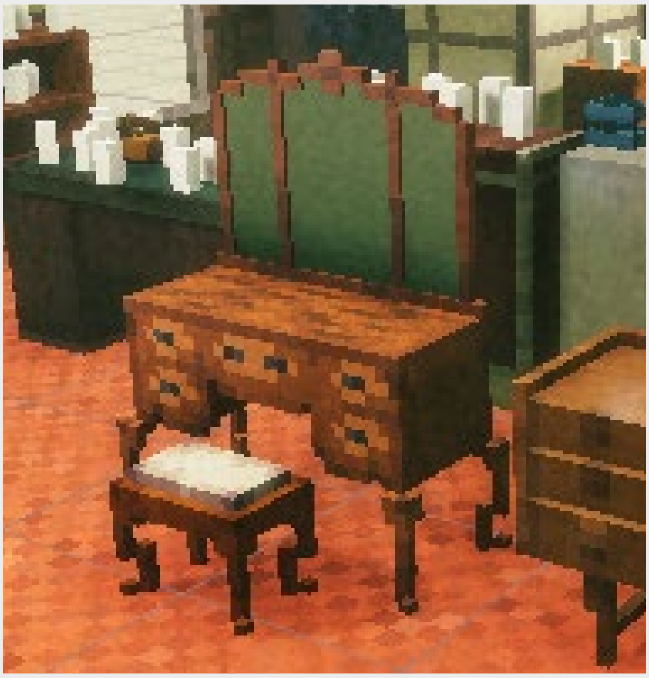


The kitchenettes are then imported into the game, and adjusted to look like so.

I also went about creating various tables for offices and bedrooms. Many of my inspirations were found on Google and Pinterest, however, most notably the prop design of classic Noir films were invaluable, such as The Maltese Falcon. The original desk that Cole has designed (the desk on the left in the image below) served as an excellent start to create my designs from, and is what influenced the design of the office chair.

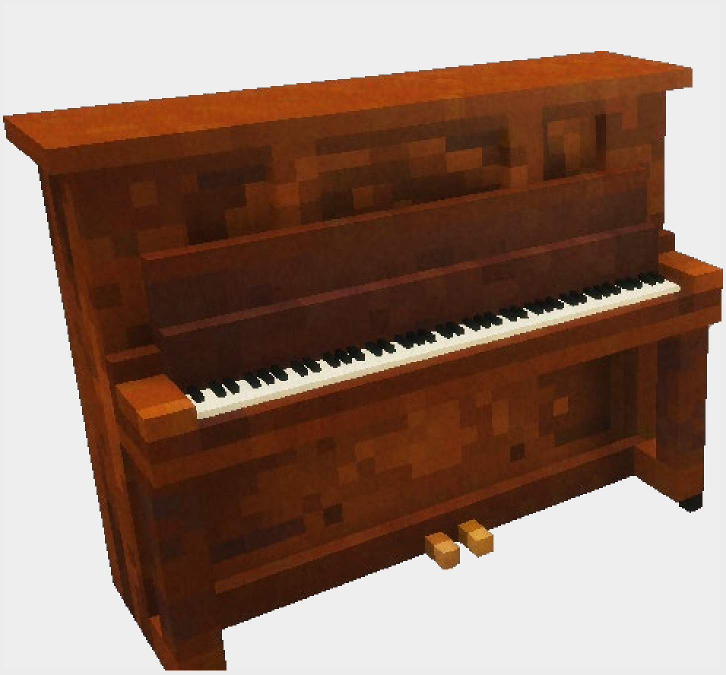


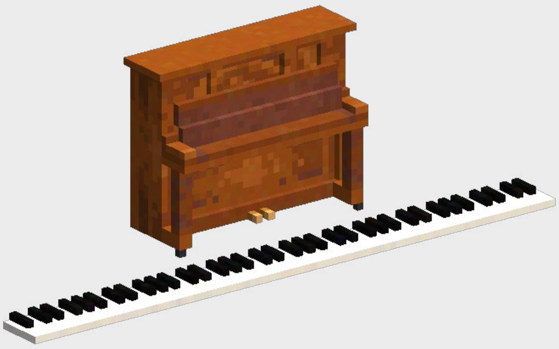
I designed some fancy vanity tables with 'mirrors', which is simply a highly reflective green gradient surface, as having a full blown mirror would be too resource intensive. Similarly, as the game is procedurally generated, having a pre-rendered mirror is not an option, so real-time reflections could incur all kinds of bugs! Moreover, the player character doesn't have a model, they're just floating arms, and so a real mirror would break immersion.



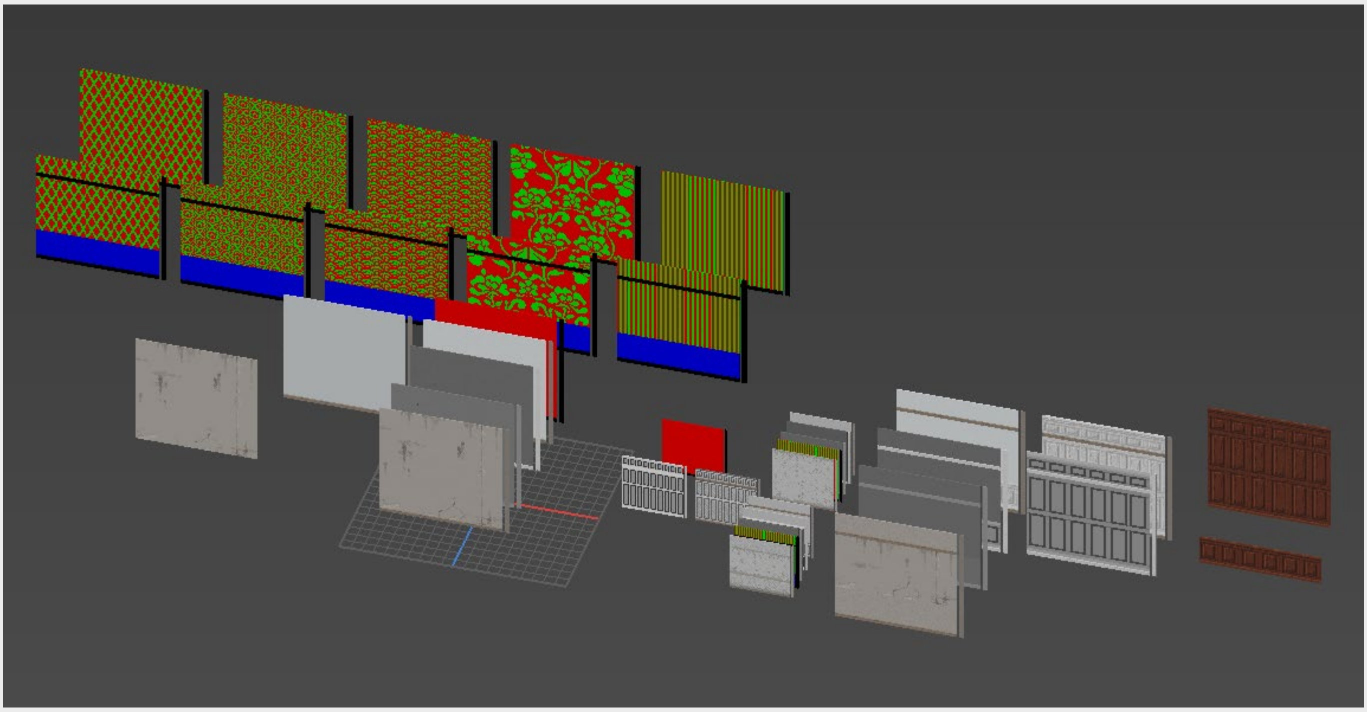
Almost all chairs and seats have a height of around 12 voxels from the floor to where the NPCs or players will sit. This is important as this is roughly the height of which the animation of sitting works without the citizens clipping into the furniture.

Next up was a Piano, creating the gritty wood aesthetic was by far one of my favourite aspects. I would mix in a little purple colour for the grains of the wood, whilst mimicking the dithering style Cole had already established with different shades of brown to get a "realistic" feel.



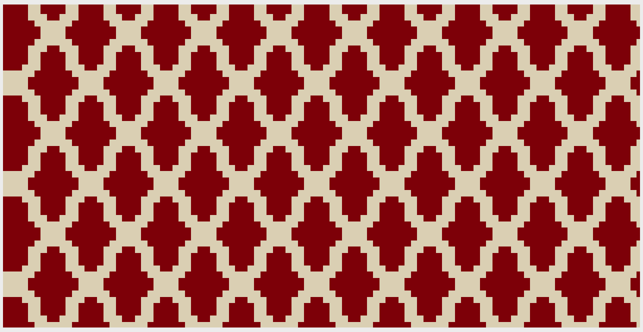


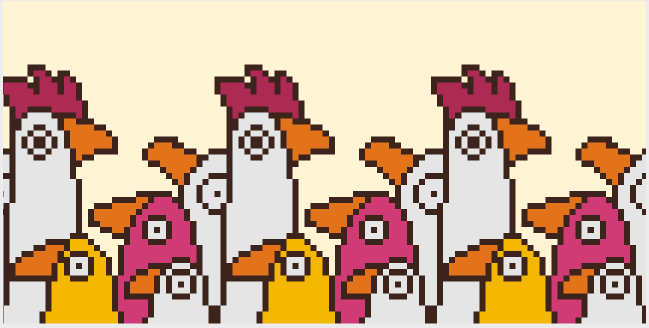
Models could also be mixed resolution, so for the piano the keys were modelled separately and then scaled down in the Unity Engine to fit on the piano body.



INCREASING WALL TEXTURE RESOLUTION

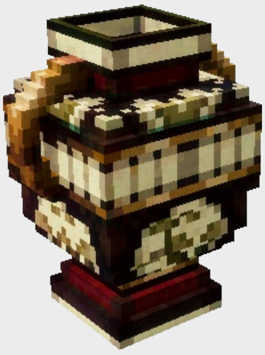
When creating furniture props for the game, their resolution was double that of the walls and floors. At some point during development, I was tasked with upscaling the resolution of said walls and floors, both inside and out to match the furniture resolution. This was an excellent opportunity to work on repeating textures, and complete the interior design. I really enjoyed refining the wood wainscoting panels to emphasise the mid-century and vintage feel of rooms, whilst contrasting this with designing the texturing for the plastic lino flooring. Working off the initial low resolution designs by Cole, reinterpreting them at the increased resolution gave the opportunity to have some really elaborate wallpapers

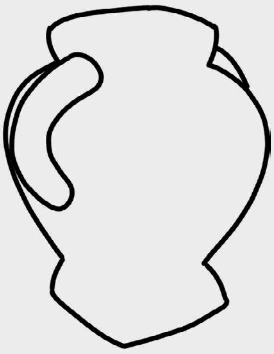




DESIGN AND INSPIRATION

Many props within Shadows are inspired by influential designers, such as Charles Eames. This was done to help place the world in the 1980s. Combining real designs with original designs in one setting creates an uncanny feeling of familiarity when walking around the city. This helps emphasise the dystopian setting.





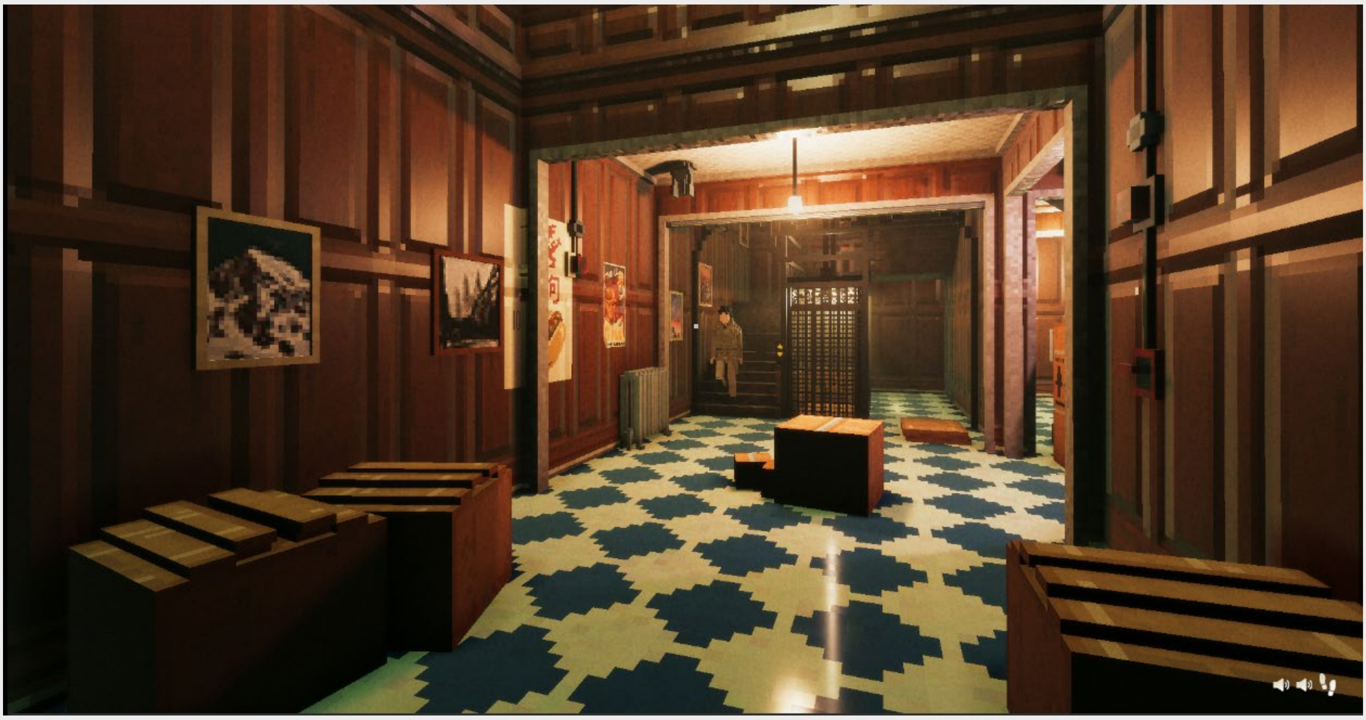
Often, assets within the designs of the rooms were inspired by set pieces from films. Such as the Maltese Falcon, The Shining, and other famous films. One of the carpets, and the sofas are directly inspired by 'The Shining'.

Even aspects of the offices were inspired by Frank Lloyd Wright's infamous Johnson Wax complex (1939). Despite this dystopian world, it's not all sardine cubicles. If anything, having a murder take place in what seems like an art deco studio straight out of 'The Shining', is probably more sinister.

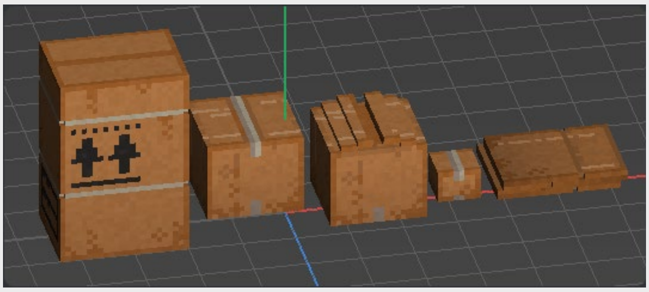
Another common place design aesthetic was art deco. Here we can see some lamps in both a vintage and art deco style. The lamps use subsurface scattering to achieve a glow on the lampshade that appears translucent. This was done because otherwise the lampshades look like a solid and opaque radioactive material that itself is emitting light. This is effective for things such as the neon lights, however.

Whilst the aesthetic of Shadows is gritty realism, it became apparent when making models that simply sculpting a curve in voxels as if it were a normal model created far too many artefacts known as 'greebles', creating this noisy and ugly surface. Instead we opted for an almost low poly look, which could then be made to fit the gritty realism look through texturing and shaders.

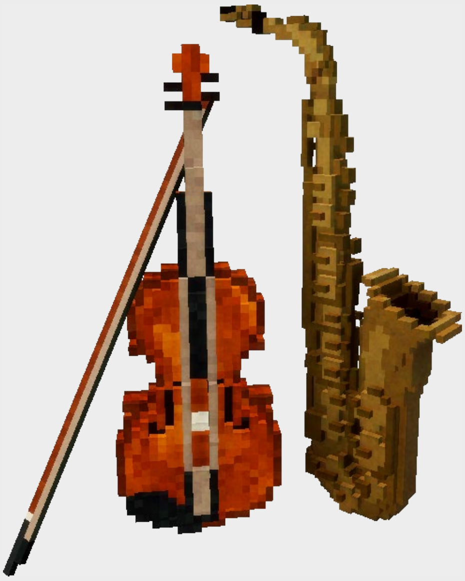
This antique vase is an excellent example. The shapes are cubes with flat faces, however the silhouette is curved, as are the handles. This gives the illusion that it could be rounded. As the real-life version of a vase is rounded, the model gives the same aesthetic as a real rounded vase, whilst having significantly reduced polycount and no ugly greebling.



Moving in day, boxes were littering the floors as new props were being imported into the game. (This wasn't intended, Cole had just set the boxes to spawn everywhere by accident).



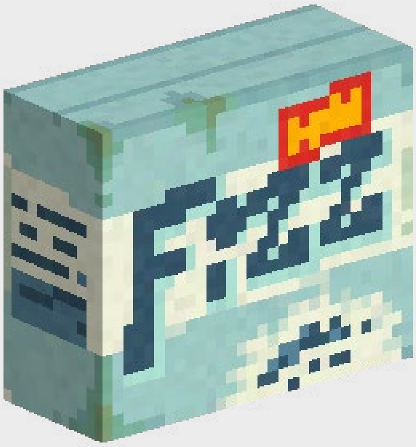
Here's a selection of various musical props.



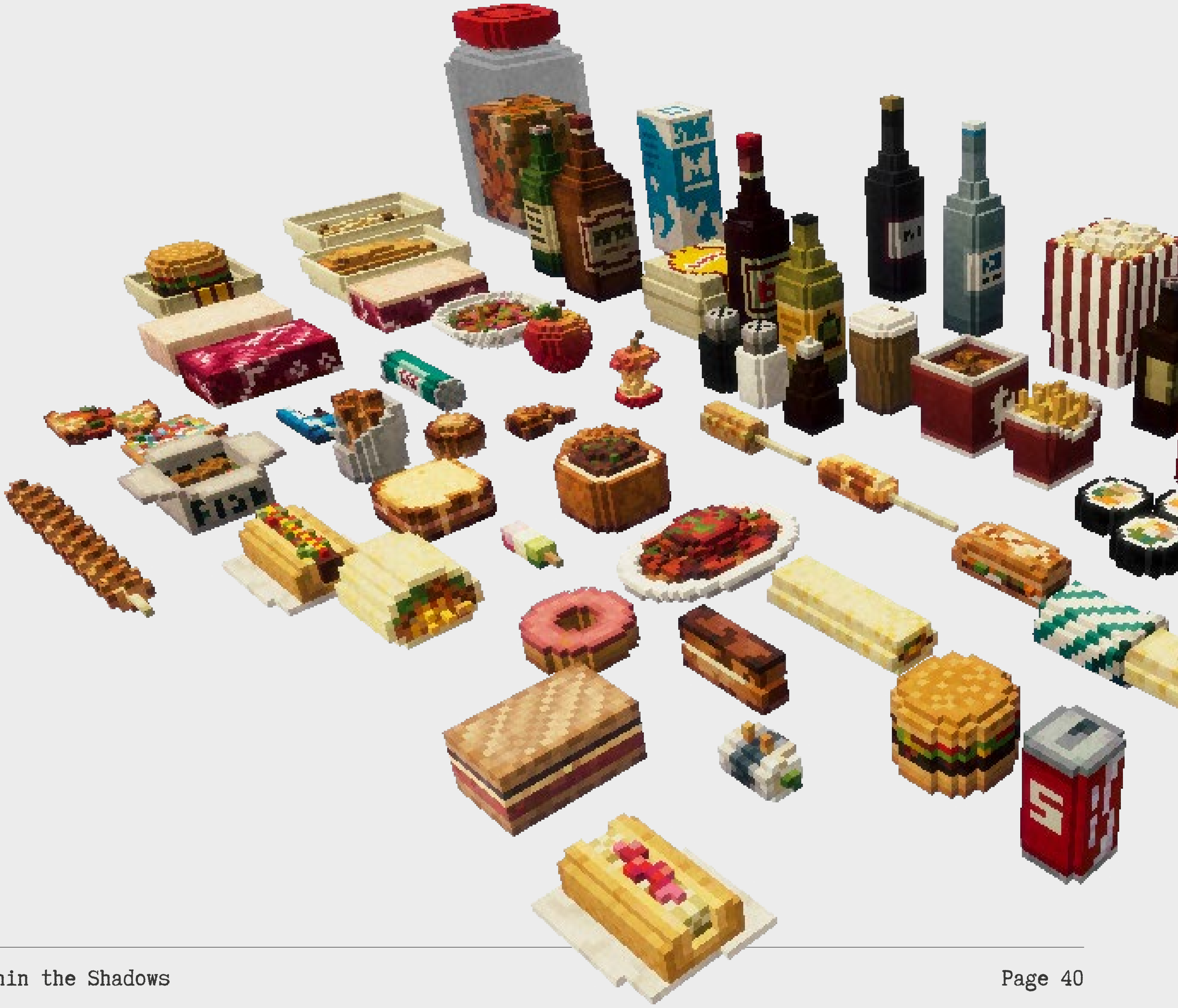
PERSONALITY ITEMS

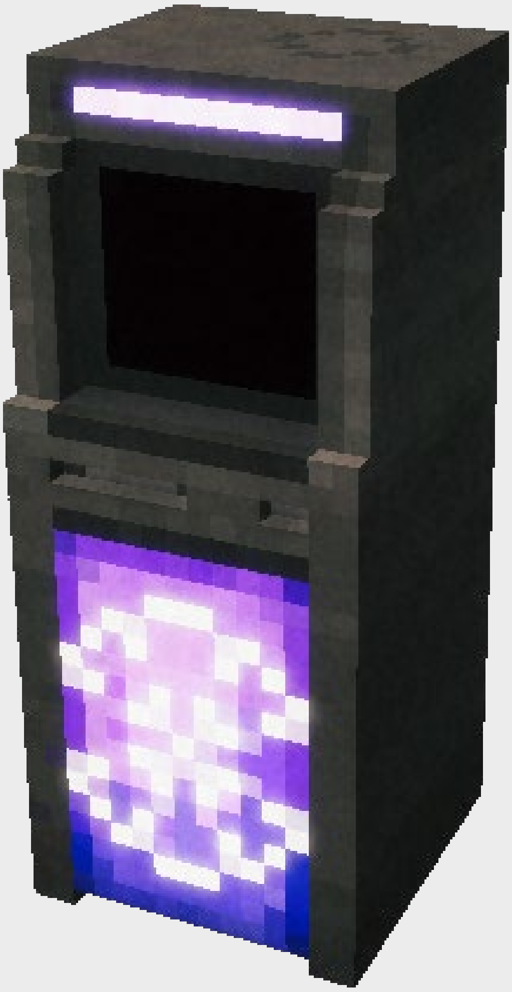
After the furniture was established, then came the evidence items. These props were either things that could be useful for a case, or simply to decorate an apartment. Factors that affect what appears in someone's apartment include; personality, wealth, gender as well as specific interests (which is determined by a citizen's personality). Therefore a wide array of items needed to be made that could be filtered by these categories. For example, an emotive or creative citizen might be interested in the violin and a piano, or if a piano doesn't fit in their apartment, or they can't afford it, just a violin.

For the process of making these 'Fizz' laundry soap pads, a brand was invented and designed off pre-existing items like 'Brillo' or 'Daz'. Dirt was added to most items to give it a grungy aesthetic that also looked 'vintage'.



There's food in the game naturally, and part of that included having eaten, and uneaten variants so the player could tell if they were consumed. Additionally some foods had eaten/wrapper variants which would then be thrown on the ground as litter - a common place feature of dystopian cities.

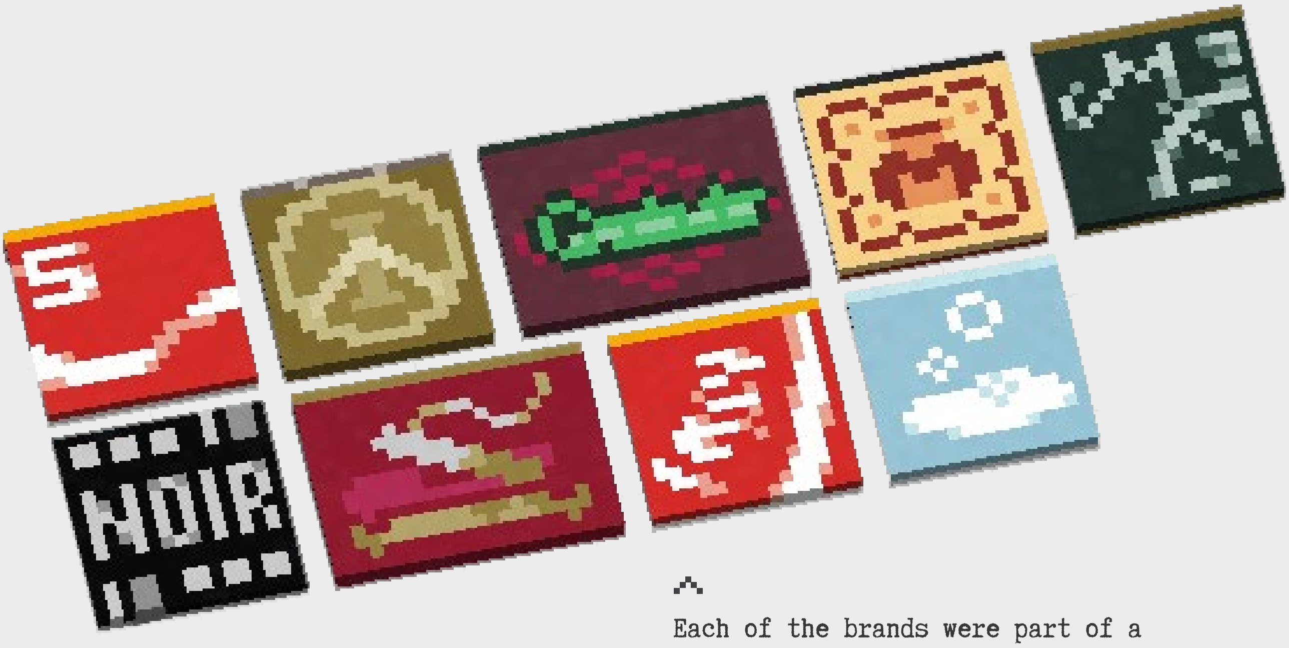




Branding and Matchbooks

To add flavour to the world, as many little details relating to the consumerist nature of the city was instilled in the props. For example, matchbooks are often used as a form of advertisement and therefore rather than a generic matchbook brand, we used preexisting brands in the game world. As well as the 'Fizz' brand I had created for the soap pads.

The ATM designed by Kensington Indigo has their colours, and a really low resolution version of their logo.



Each of the brands were part of a wider conglomerate.

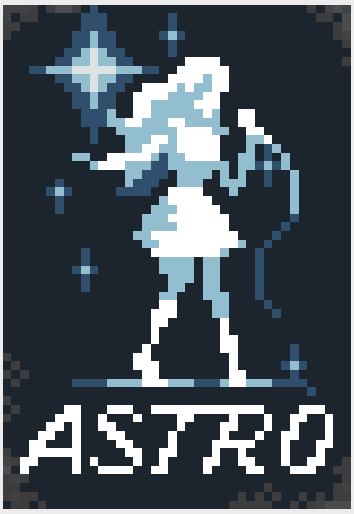
Film Posters

With film noir being an important inspiration for the project, film posters were an essential addition to the game. Stark (Holborn, Lead Writer) had already come up with numerous titles and therefore it was up to me to interpret these! Using classic detective film posters as inspiration here's some of what I came up with.

Also more adverts!

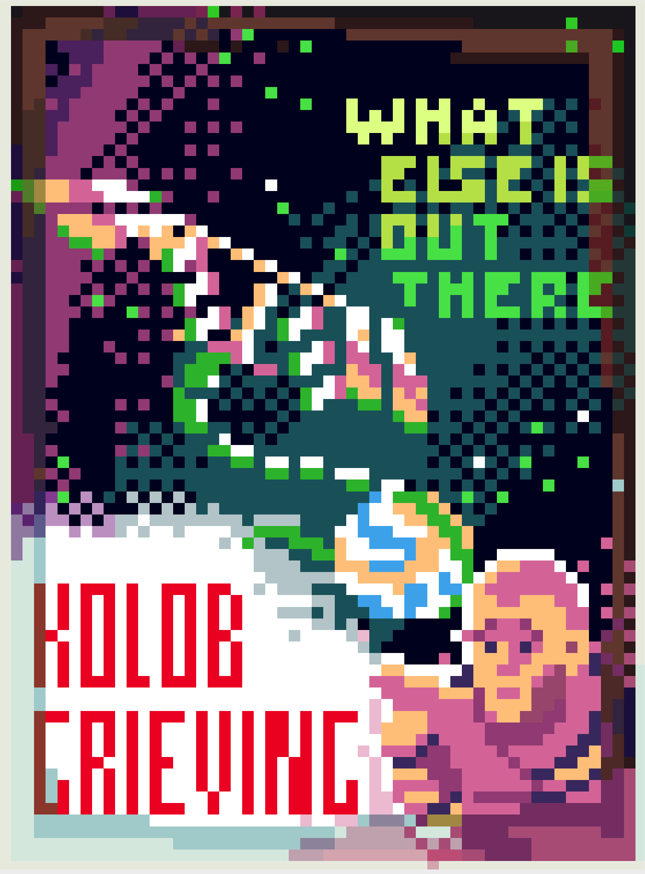
Gemsteader

THRILL
GILL



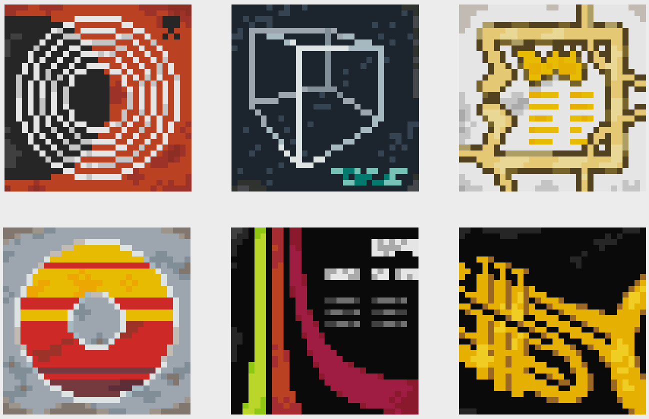
ELLIOT
EG
GENETICS

A poster of the Cosmic Pop Princess, Didi Souris. Musique Astro! is hugely popular with younger listeners, and is the new face of Elliot Genetics.



For this poster a vintage sci-fi colour palette was used, this imagery based off this piece of text written by Stark Holborn:

> Kolob Grieving
>
> "Ten years after leaving Earth, as the solo engineer of the first astro V-Mail pipe, Kolob King waits alone on a satellite station for the message that will announce the first wave of settlers are on their way. However, when a message finally arrives, it is not a greeting but a random page from novel, seemingly about his own life. As more pages arrive, out of sync, Kolob begins to wonder what is left on the other end of the tube..."

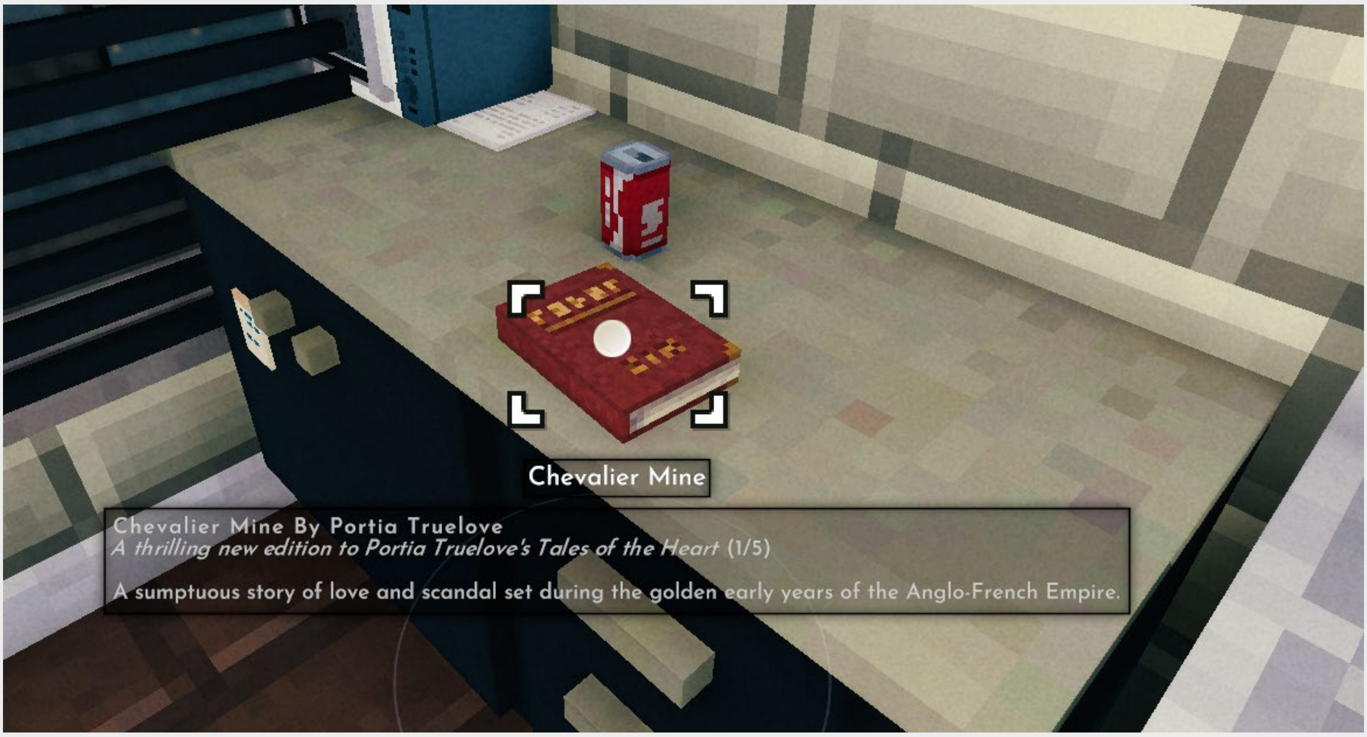


Vinyl Covers

The designs for these vinyl covers were inspired by the various ideas Stark Holborn had come up with.

Books

With the brilliant writing of Stark Holborn in the game, we needed some beautiful book covers to go with it. These were higher resolution than the other books on the shelves, to show to the player that they were interactable. Meanwhile, the shelves themselves were modelled with spaces in between the books for these books to spawn in randomly.



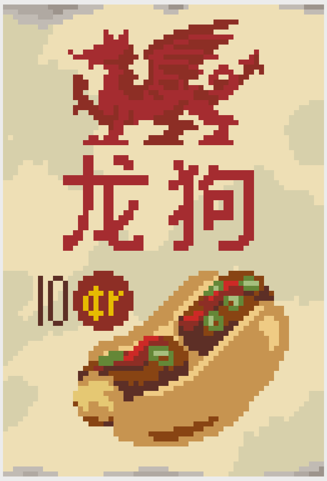
Chevalier Mine

Chevalier Mine By Portia Truelove

A thrilling new edition to Portia Truelove's Tales of the Heart (1/5)

A sumptuous story of love and scandal set during the golden early years of the Anglo-French Empire.

Tornado Potato, a popular street food in Korea.



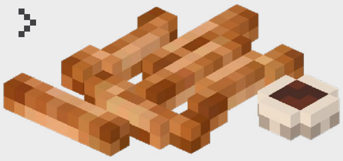
Cultural Items

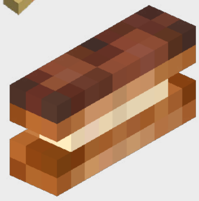
To build out this world and really sell it as something unique, specific cultural environmental storytelling was used. Posters are an excellent way to do this, there's frequently empty walls, and store fronts that could be plastered with these things without looking out of place and repetitive.

The cultural mixing in Shadows has resulted in these amalgamations of classic food dishes, such as the Welsh-Chinese 'dragon dog'. Additionally, placing a pricing on these with the crows currency symbol would help ground the player in this fictional alternative timeline.

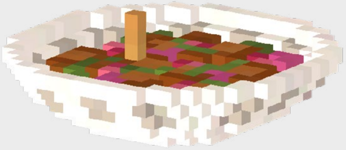
Kushi Dango, a sweet Japanese dumpling.

Churros, a sweet Spanish and Portuguese treat.

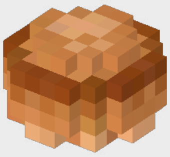




Chocolate Eclair, a French pastry.

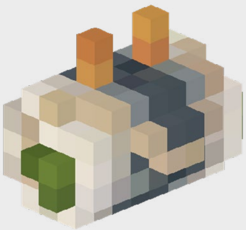


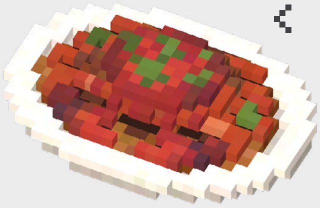
Aloo Chaat, a street food dish originating from the Indian subcontinent.



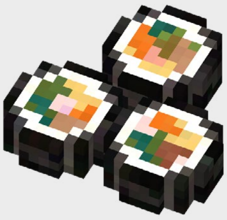
Yorkie Pie, An enclosed Yorkshire Pudding with filling of meat, gravy and potatoes. Popular in the old days of the empire. (A Shadows of Doubt original.)

Rollmop, pickled herring fillets from Germany.





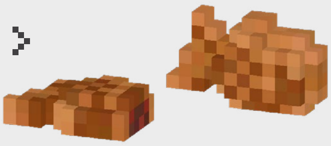
Chilicrab a popular dish from Singapore and Malaysia.

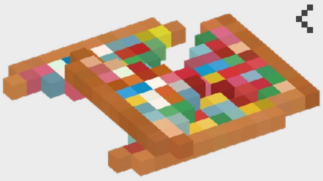


Gimbap a Korean dish with rice, vegetables, fish, and meat, rolled in seaweed.

Bánh Mì is a baguette like sandwich with savoury fillings, it is a part of vietnamese cuisine.

Bungeoppang a Korean fish-shaped pastry.



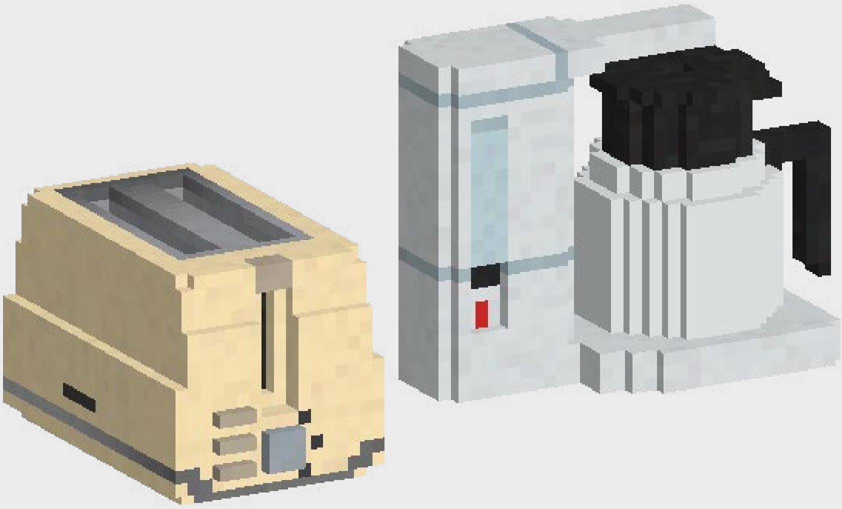


Fairy Bread a traditional Australian snack.

This was a post for thanksgiving - supposed to be a happy family occasion, instead this family is full of discord and processed foods.

KITCHEN APPLIANCES

For the kitchen appliances, I wanted to make sure they were mundanely retro. With an insipidly beige plastic covering and bulky form factor for the toaster. All to highlight the dreary dystopia of the cities.

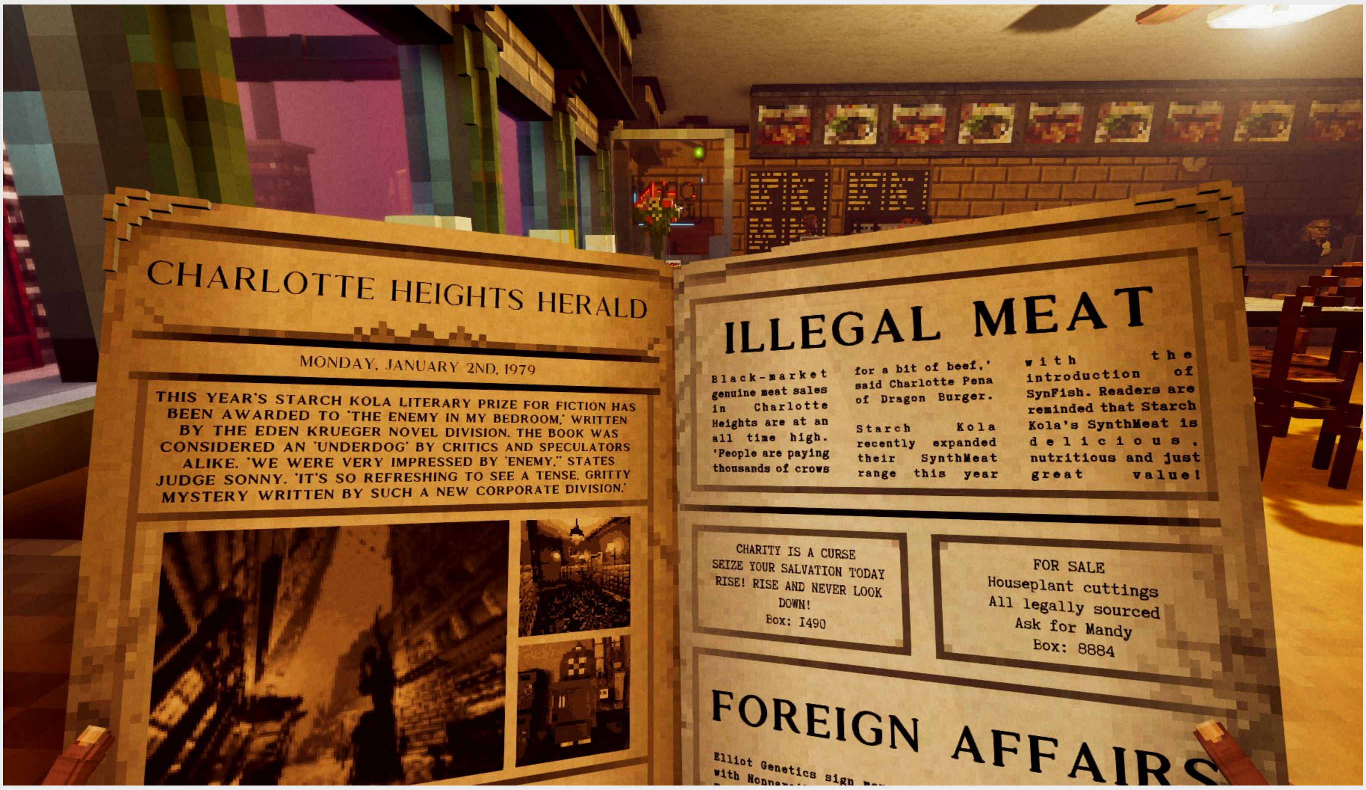


DETECTIVE EQUIPMENT

The first person models were a great opportunity to get really absorbed into the details of objects. The player first gets introduced to their basic detective gadgets near the beginning of the introductory mission; a print scanner for identifying finger and foot prints, along with some handcuffs and some lock-picks. Later on in the game you may find more gadgets to expand your abilities; stun grenades, codebreakers and even a tracking device.

Newspaper

The newspaper first design. Many props have two model varieties, one that is in world and one that is the first person model. For the newspaper the idea was to have one that players can hide behind while sitting in diners... maybe some eye holes to peek out of too! The newspaper now functions as a in world record of the events that have unfolded (pun not intended), when you solve crimes or a murder is committed.



CHARLOTTE HEIGHTS HERALD

MONDAY, JANUARY 2ND, 1979

THIS YEAR'S STARCH KOLA LITERARY PRIZE FOR FICTION HAS BEEN AWARDED TO 'THE ENEMY IN MY BEDROOM,' WRITTEN BY THE EDEN KRUEGER NOVEL DIVISION. THE BOOK WAS CONSIDERED AN 'UNDERDOG' BY CRITICS AND SPECULATORS ALIKE. 'WE WERE VERY IMPRESSED BY 'ENEMY,'' STATES JUDGE SONNY. 'IT'S SO REFRESHING TO SEE A TENSE, GRITTY MYSTERY WRITTEN BY SUCH A NEW CORPORATE DIVISION.'

ILLEGAL MEAT

Black-market genuine meat sales in Charlotte Heights are at an all time high. 'People are paying thousands of crows for a bit of beef,' said Charlotte Pena of Dragon Burger.

Starch Kola recently expanded their SynthMeat range this year with the introduction of SynFish. Readers are reminded that Starch Kola's SynthMeat is delicious, nutritious and just great value!

CHARITY IS A CURSE
SEIZE YOUR SALVATION TODAY
RISE! RISE AND NEVER LOOK DOWN!
Box: I490

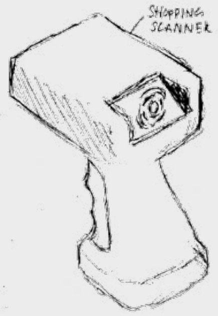
FOR SALE
Houseplant cuttings
All legally sourced
Ask for Mandy
Box: 8884

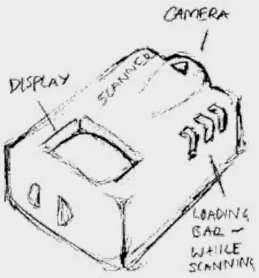
FOREIGN AFFAIRS

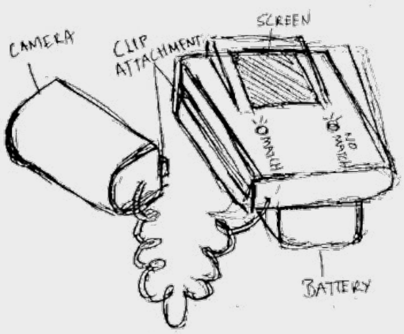
Elliot Genetics sign
with

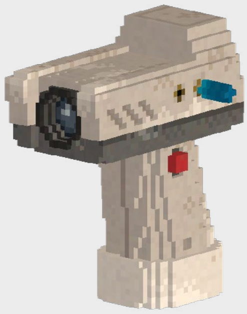
Technology

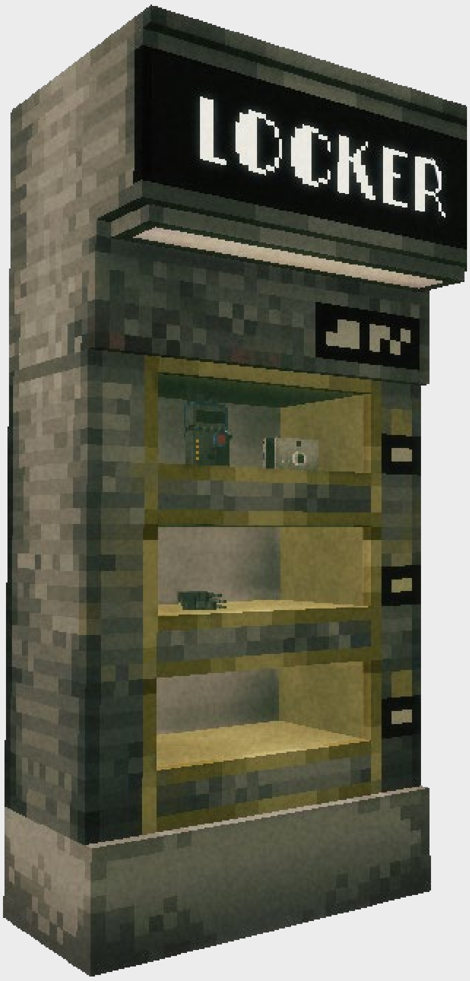
The design of the technology in the game is beautifully obtuse, where function trumps form. Whilst a sense of ergonomic form is attempted, there are clear limitations with the design, creating these bulky pieces of machinery.





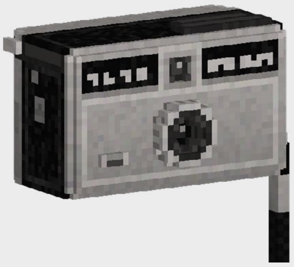






The designs for the fingerprint scanner can be seen below. A mix of gameboy style designs and the final "shopping scanner", based on the barcode readers you find at supermarkets. The coiled wire was also added to this final design that you now see in game.

The lockers are how the police obtain their weapons and technology.



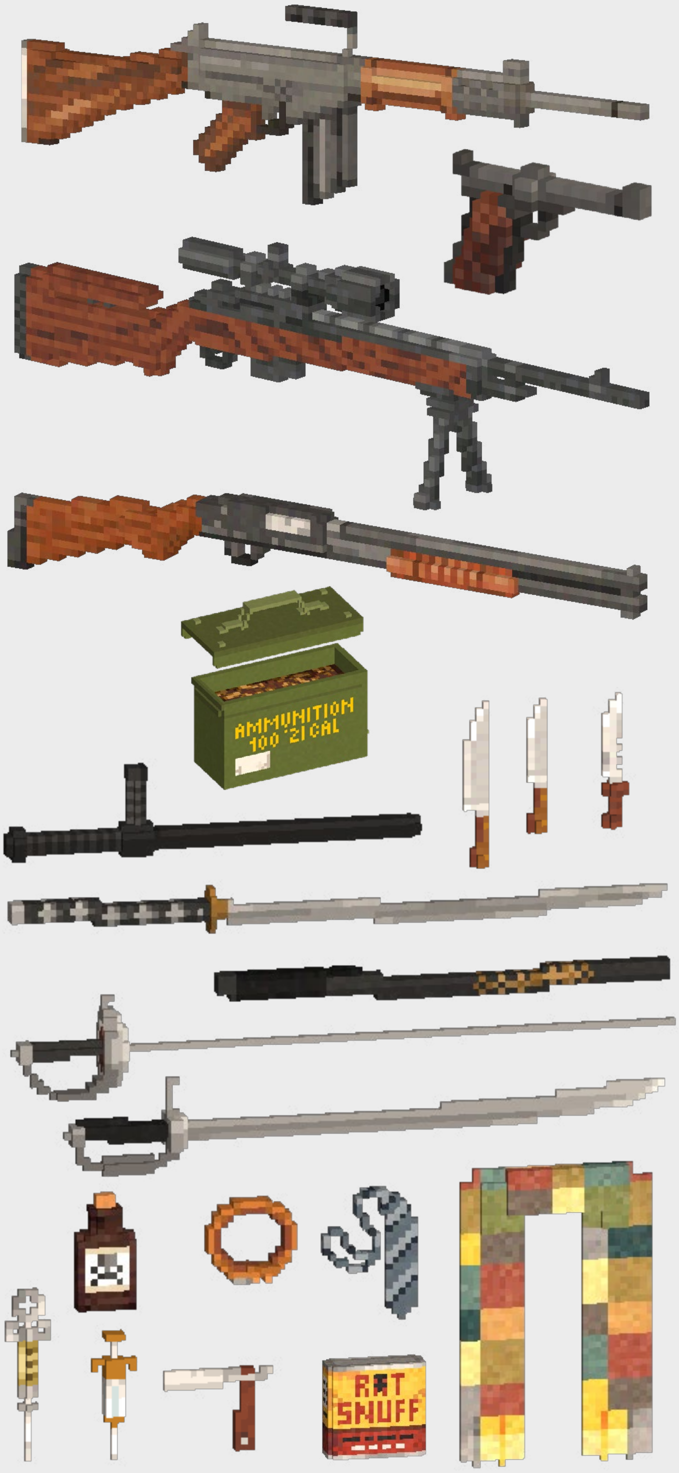
GUNS AND MURDER WEAPONS

Guns were to be added simply as in world props, finding the right kind of gun so then also make it feels slightly techy. The type of guns I used as inspiration for the modelling were those which could be easily accessible to the public; those that were maybe left over from wars. In addition to this, aesthetically, guns with a bit of wood, but also a somewhat "techy" appearance to them.

The rifle needed to come with a stand, while this was modelled in the summer of 2021, Cole was already talking about the sharpshooter update. This wasn't to be a gun that could be picked up and shot. An early design choice meant that no guns could be fired by the player anyway. This would instead be a piece of evidence to be left behind by a killer from a building across the way from the victim's apartment.

So much detail was put into the props in this game, there's a documented scale used for export settings and the general scale of voxel models. As such this was used as a reference point to get the specific mm scaling of the bullets to then be found in game as evidence.

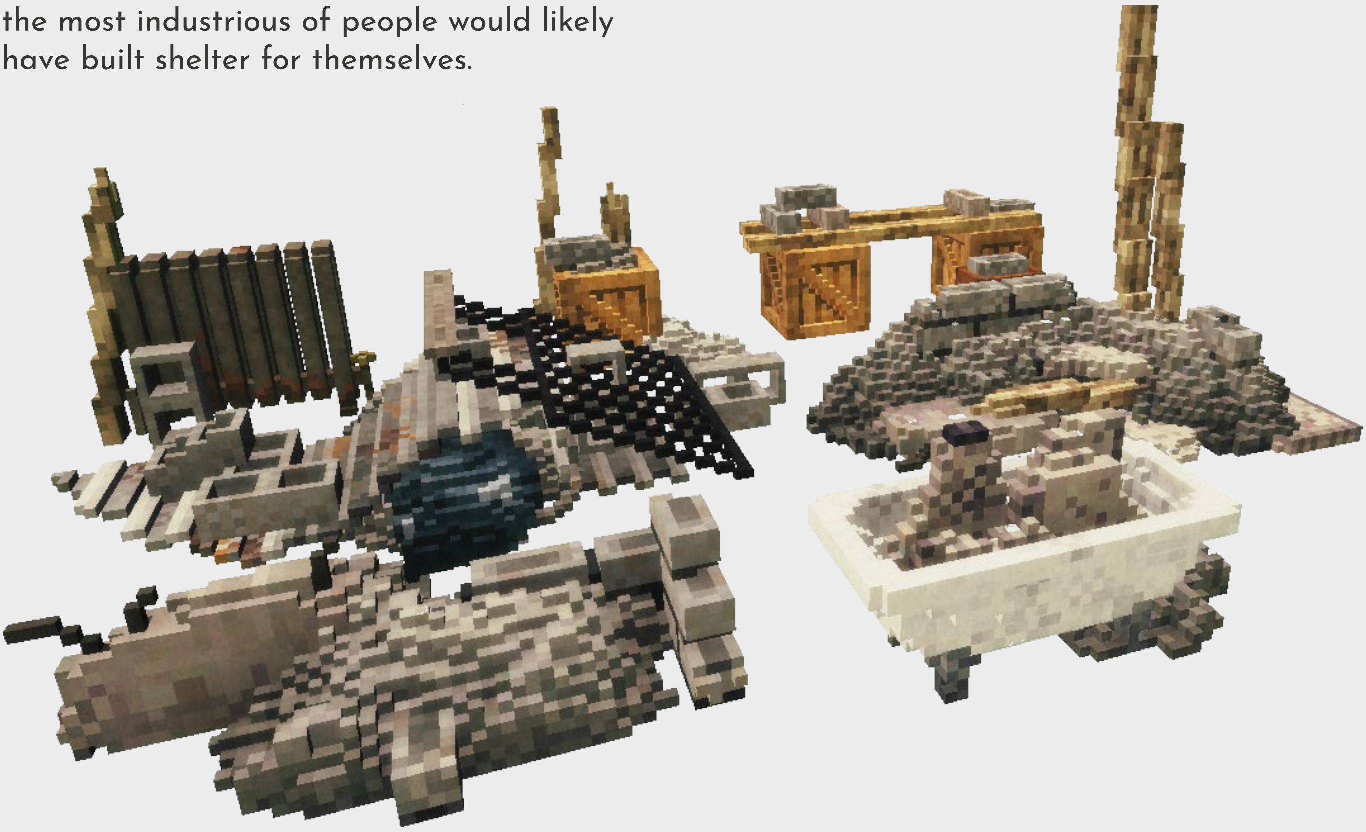
As is often sadly the case in game dev, not all our murder weapons made it to the final game: The main culprits being the softer/flexible items you see here, which are incredibly difficult to animate properly in voxels!



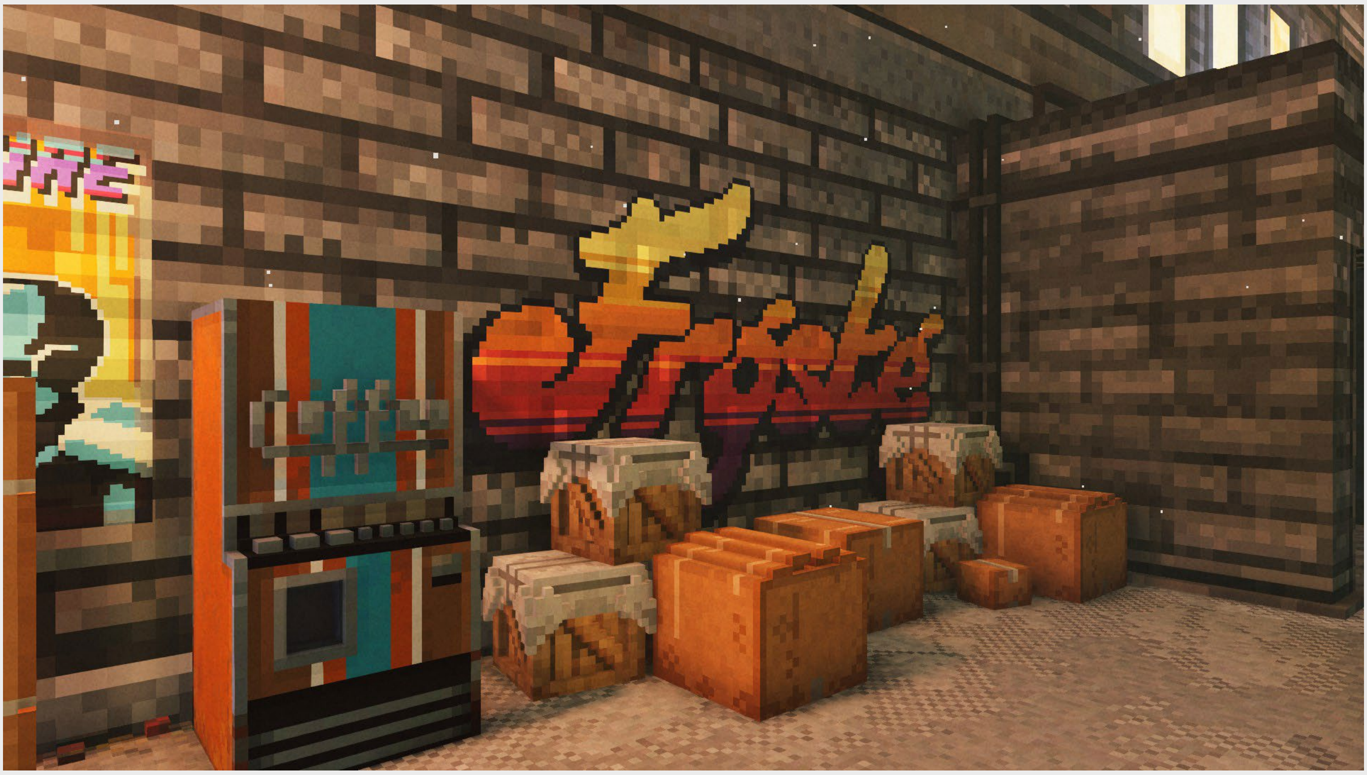
Murder weapons! There's so many to choose from; from the old trusty kitchen knife, to the conniving rat poison, and the slow sadistic belts, ties and scarves.
As is often the case in game dev, some of the above didn't make it into the final game.

THE STREETS

Once the interiors were starting to fill up and there was at least one object of each type for each design style, I moved outdoors to decorate the streets. Homelessness is rampant within Shadows of Doubt, however the most industrious of people would likely have built shelter for themselves.

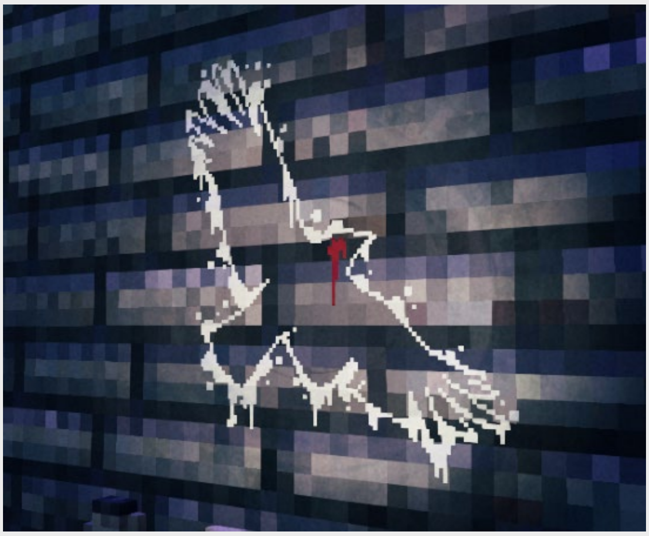


To add texture to the streets, I created some rubble. One of the biggest limitations of Qubicle was not being able to 'sculpt' forms, which I later started using Blender to do. Instead you have to either make large volumes and then delete bits, voxel by voxel, or build up from single voxels. However, within Qubicle what I did discover was that you could create terrain maps and this became the start of a way for me to not only make interesting forms, but also better textures for large areas. Once I had a "rubble pile" I added weathered and broken versions of other props to create these junk piles.



Graffiti

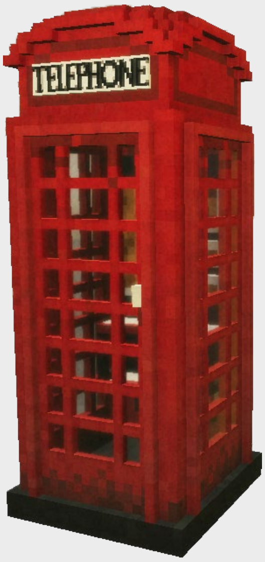
Going back to the idea of decorating walls with easily repeated assets - graffiti was an important decal to include. As an easter egg I added the developers names as graffiti tags. Stark had some brilliant ideas for the specific faction world building elements, creating prompts for social and political graffiti in the game, as well as referencing significant films that helped inspire shadows.

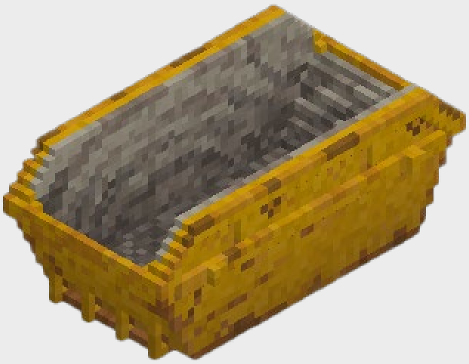


Street Furniture

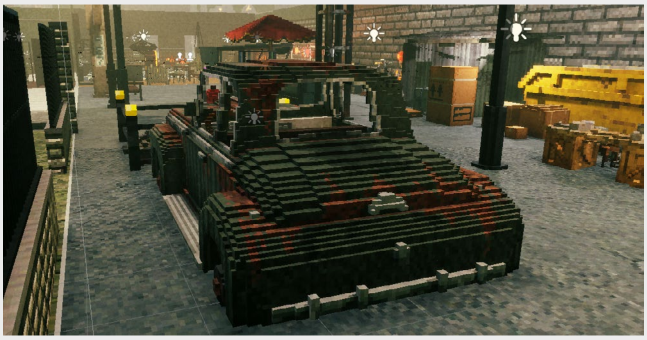
The telephone boxes were given a heading with the same font as traditional British phone boxes, at a much higher resolution to the body of the model which would then be scaled down.

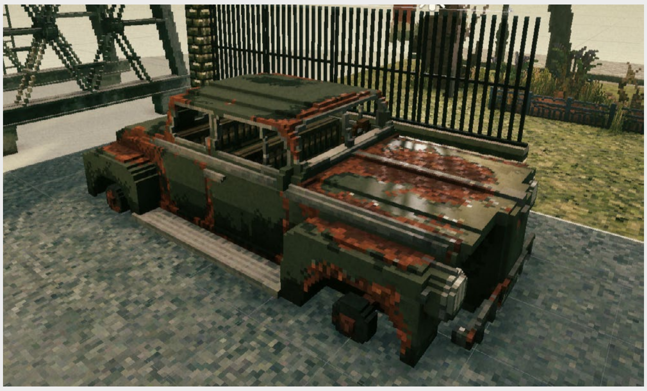
As Shadows of Doubt has such a unique art style it was an excellent opportunity to play around with the process. For this skip on the inside I tried using swirling lines to inform the dirt pattern, somewhat influenced by Van Gogh's style. I know, an odd choice of object for this, but with such a flat grey area it felt like a fun piece to try to make a bit more interesting. I kept using a very similar dirt texturing technique for other objects. This painterly style came to inform other designs for the later in the project.

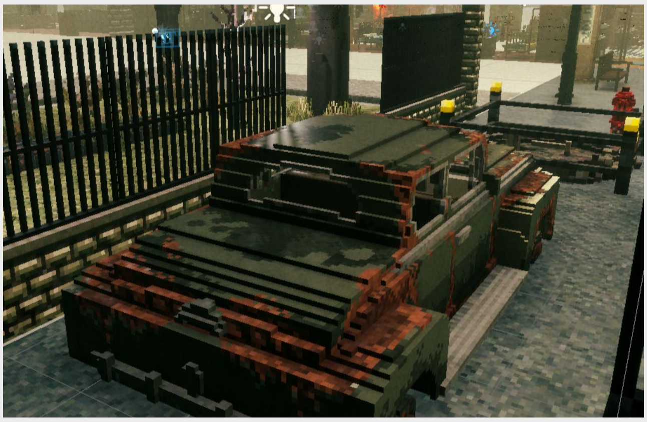




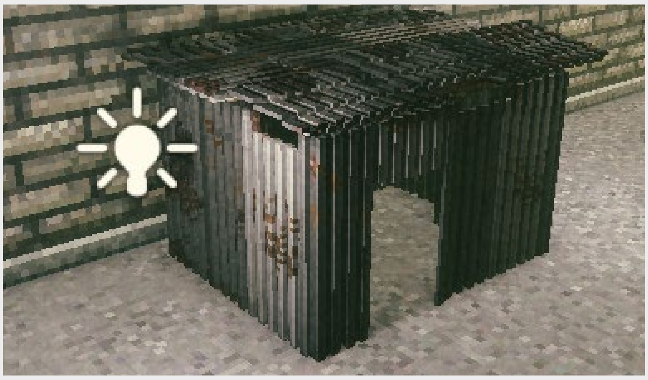
This image here really helped captivate a feeling of being off the main street. Which up until now was spotless.

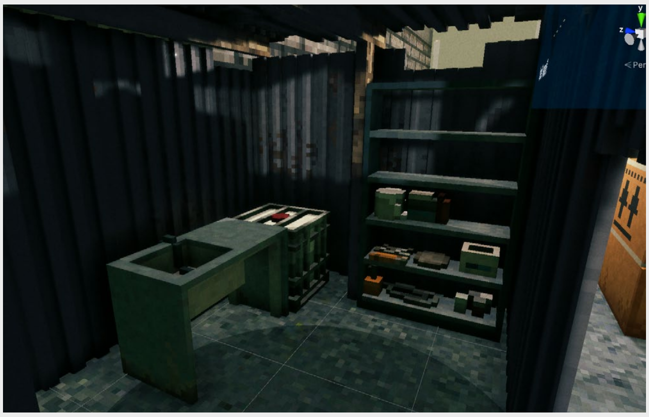


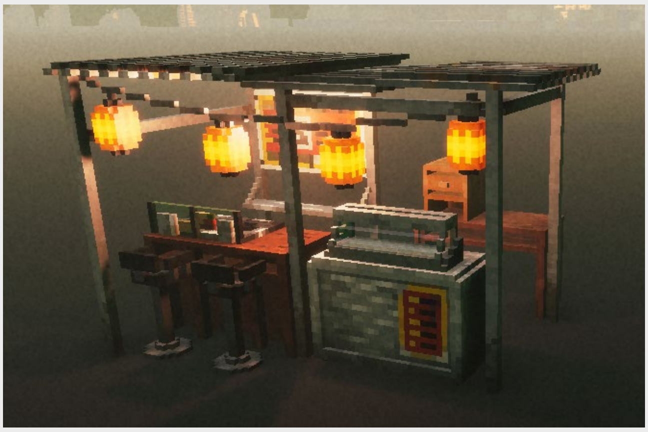




The car had way too many greebles, much like the bleach bottles. Therefore to try and remove this, I had to remove the beautiful curves of the car and make it square.



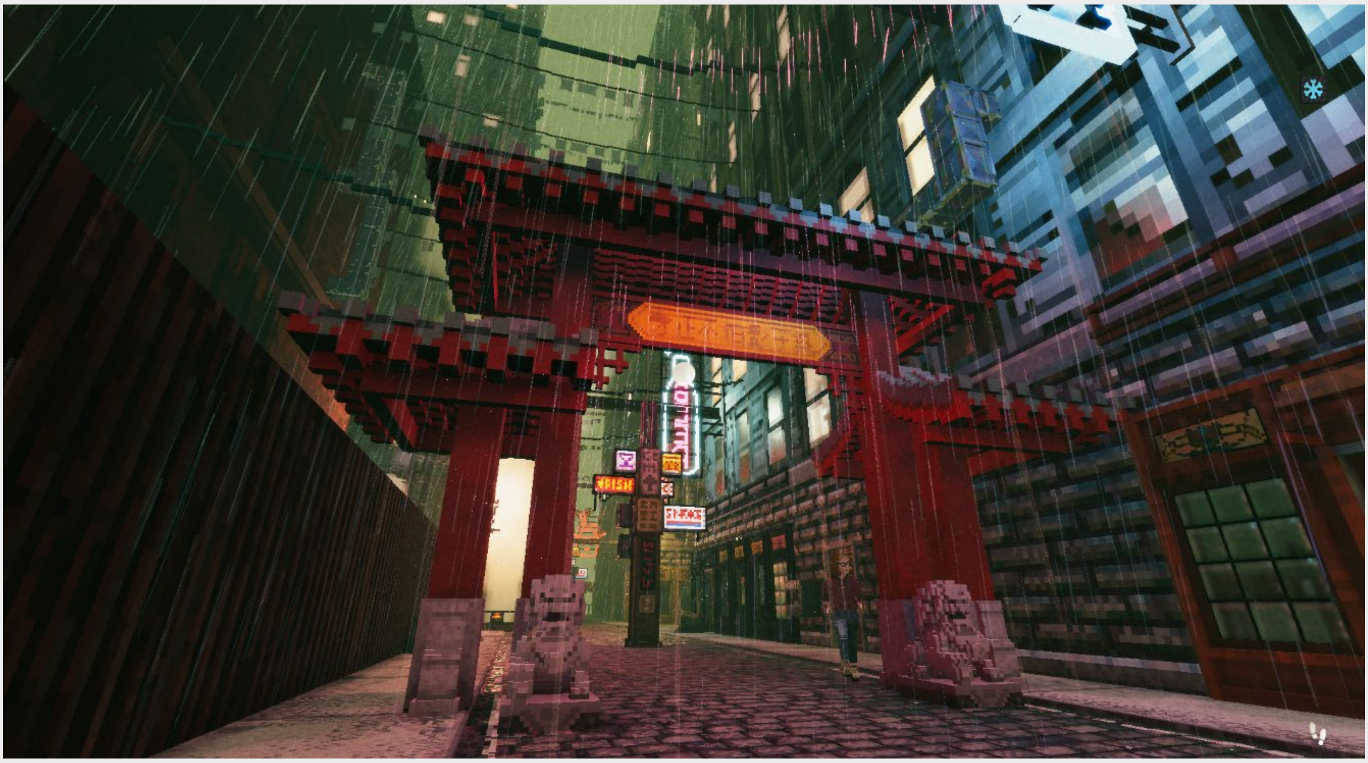




The process for creating the shanty shack was relatively simple, however, a few different versions of corrugated iron were created to do this. I took a flat plane of voxels and for every 2 voxels offset their position by one for the corrugated form. I then coloured the lower strips a dark grey and the raised strips a light grey. Splashed on a grey gradient for that metallic sheen and then weathered it with rust. Stacking multiple of these together over a wooden frame, cutting a hole in one sheet for a door, and an effective shanty shack was made!

This was then filled with outdoor variations of the kitchenette - which contained a sink with a water butt attached. Simple bedrolls were also added for people to have some protection against the cold hard flagstones.

The corrugated iron was then reused for the food stand model. A lively little shack designed for street food sellers, with a chain of Chinese lanterns, a menu board in the back, a custom wok hob, heated display and custom high seats for visiting customers.



Paifang

In order to mark the entrance and exit of Chinatown we introduced a Paifang. Many Paifangs have a saying written on them, such as the one in Manchester which reads;

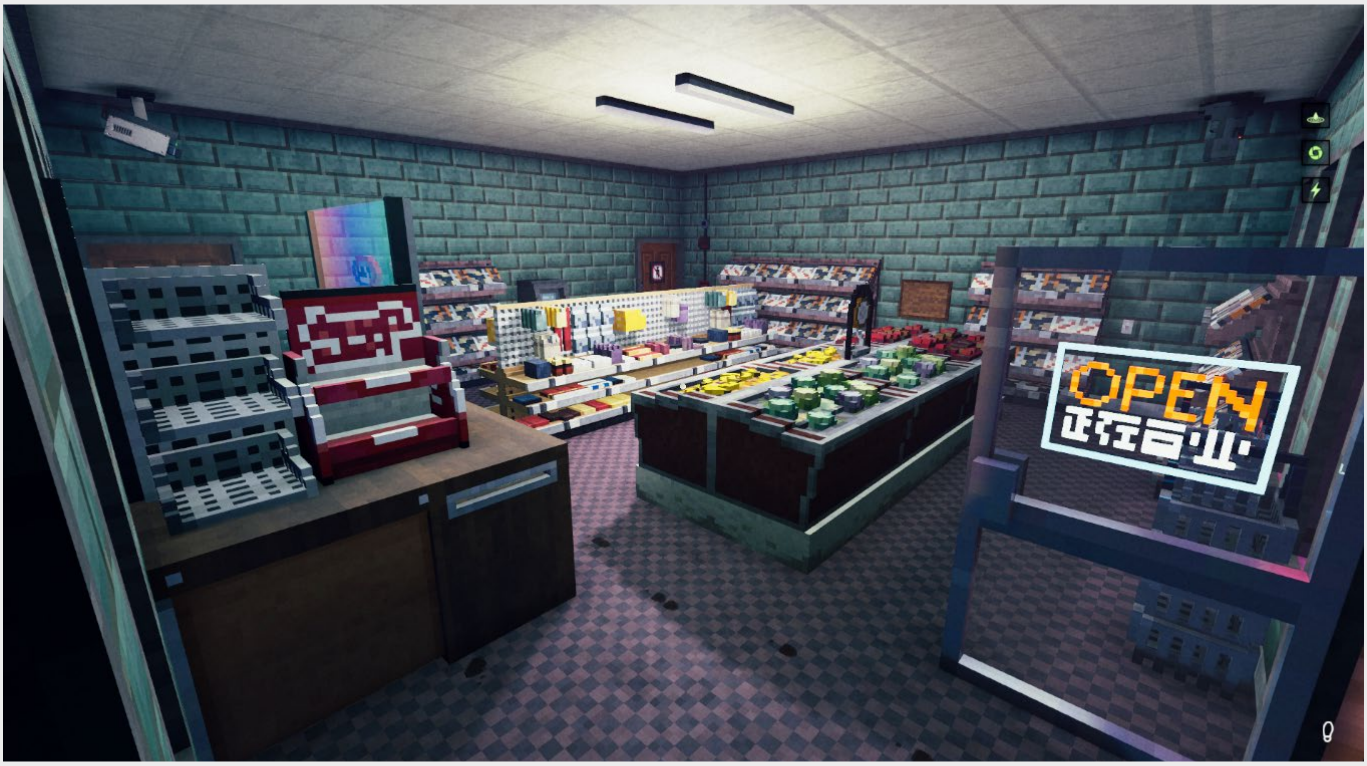
天庆地祥

This translates to "Heavenly Celebration and Earthly Blessing."

For our Paifang the Chinese proverb is written;

身正不怕影子斜。(Shēnzhèng búpà yǐngzi xié. 'body straight not fear shadow slanting') – One who stands straight doesn't fear a crooked shadow.

The references to a 'shadow' as in the title of the game fits very well, but most importantly the message suits the theme of the game very well. It's somewhat ironic, this positive message juxtaposes the tone and nature of significant actors in the game's world (corrupt companies and murderers). However, in some sense it is also calling to the player that they should not fear these 'actors' if they are standing straight, doing the right thing and finding justice for the victims of the city. How much of that is necessarily the case when most of you are breaking into everyone's' apartments, looting everything in sight!



Vending Machines and Shops

For the shops, a modular approach was taken, so that the blocks of freezers units could be procedurally generated to fit the various sizes of convenience store without the need of making several custom variations. It also simply makes sense as the shelving within real life stores are modular anyway!

Coffee
EG
HOT

Vending machines

This cash desk was created during COVID when shops were installing plastic panels in front of cashiers. However, for Shadows, it worked nicely as a bullet proof shield.

Cranes

The cranes in the game are based on the real life electric cranes found in Bristol, where ColePowered Games is located. They were built in the 1950s by Stothert & Pitt of Bath, and fit perfectly into the setting of the game. It took 3 days to model and texture this crane alone.

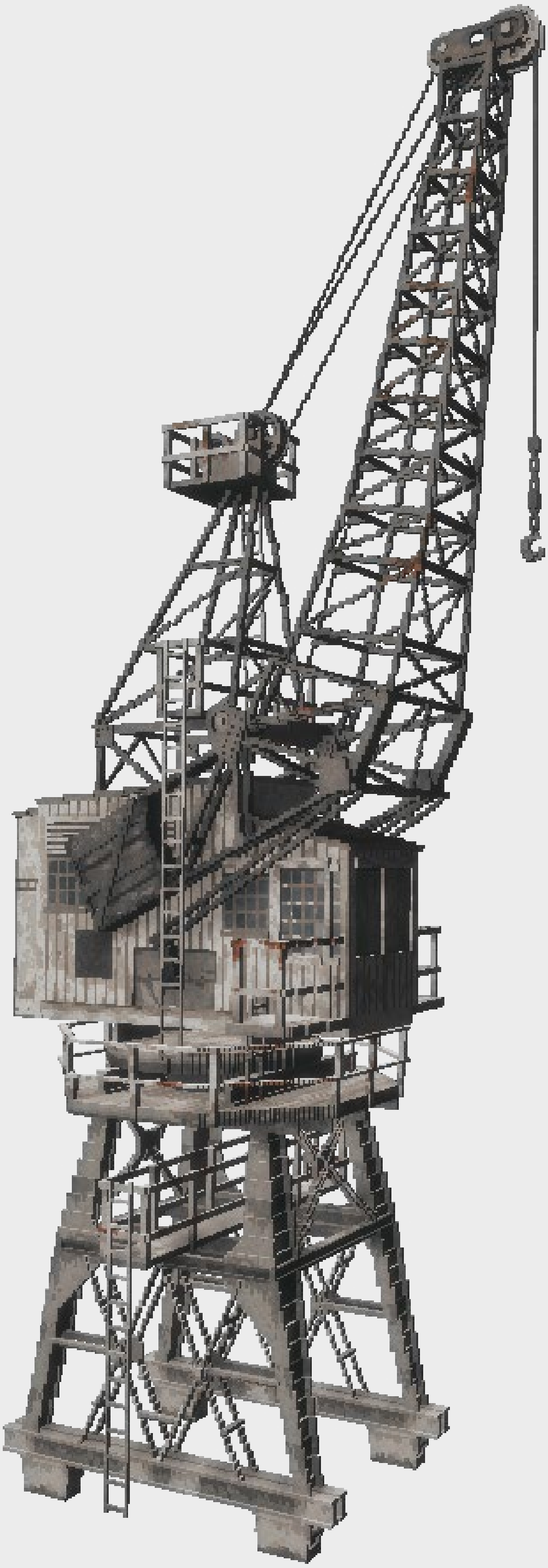
At one point it was considered to have the crane be a possible setting for a murder and so the scale of the upper floor was designed to be hollowed out and made into a room filled with evidence! It would either have been reached by a ladder or scaffolding. However, this didn't fit within the scope of the game and was left to add flavour to the hyper industrial design of the world.

Images of the process of the crane's development, starting out as a greyscale model. This was by far one of the most fun props to create - to delicately design the frame of the boom accurately from images I found online.

This crane was so large just to see the full extent of the model rendered in Qubicle I had to take a composite image! As you can see here, I used the noise mapping built into Qubicle to try to add some rust to the cranes.

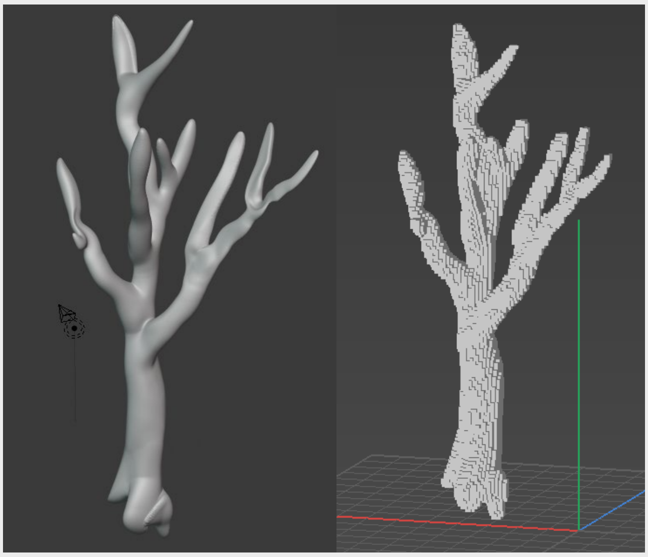
For the textured walls of the crane I used a noise map, which I then applied the game's colour palette to, and then cut and paste it onto the greyscale model.

The final completed crane model as rendered in game.



PARKS

For the design of the park at first park benches were added. However, after this the textures for the grass and paths needed to be increased in resolution, and so inspired by John Atkinson Grimshaw I created an early concept sketch to get an idea of what environmental props could be added. For this I was particularly influenced by 'Woman on a path by a cottage, 1882', with its bare branched trees silhouetting a moody blue-grey sky.

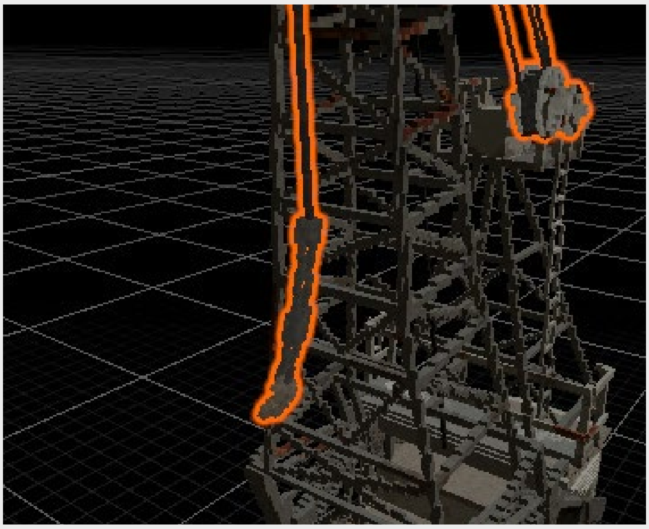


Here you can see how I started to incorporate Blender into the workflow. Starting with a rough clay sculpt of a tree and using the voxeliser in Qubicle to import the sculpt. With some heavy tweaks to the blender model, and then further refinements in Qubicle, I developed a new workflow for organic structures. This meant multiple tree variations could be created quickly and then repeated throughout the game without looking too repetitive.

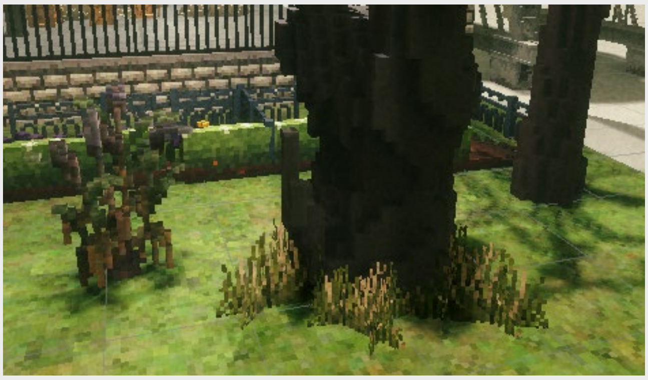
To further add life to these trees I set about creating a vertex vegetation shader to allow them to sway in the wind!

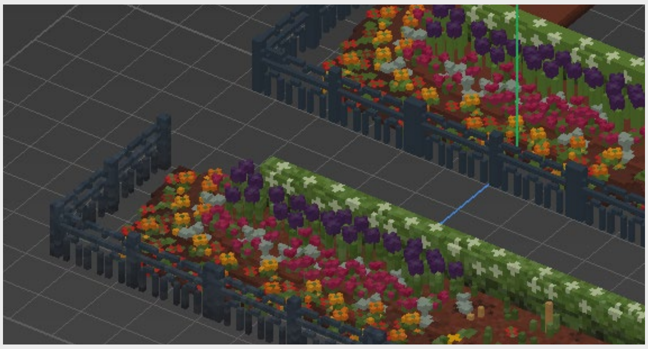
Which took a little trial and error! One of the issues with the vertex shader was that the vertices of the voxels were splitting apart. I had to tinker with various methods to make sure everything was kept as one solid swaying form.

And once this shader was created, it could also be applied to many other environmental structures which might be affected by the wind, such as the crane hook and line.



To help sell the run down feeling, Buddleias were added, which typically grow in disturbed urban environments. However, I made sure to remove any of the usually purple vibrancy from these shrubs. Halfway down, the leaves are drooping and brown to really sell this feeling.





Flower beds were made modular so they could be used to create larger areas of negative space which would break-up the shape of the park.

At first for the grass I tried billboarded sprites, where a flat 2D plane always faces the camera. However that felt weird and instead used the trick from games like Minecraft, where there are multiple 2D planes crossed over perpendicular to each other, so that at any given angle the player always sees a flat face. This grass could also be placed around the bottom of trees to help ground them, instead of looking stuck on.

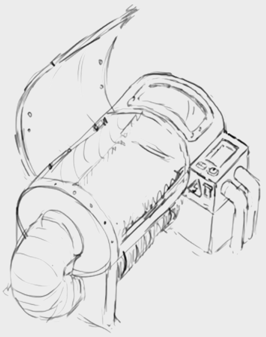
Two variations of flower bed were made which could be tiled together - luscious flower beds full of tulips and other garden flowers, and more trampled ones, where there were flowers there at some point and now there's just weeds growing out which aren't being maintained.

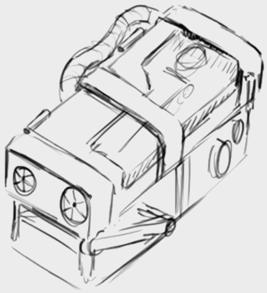
For Halloween I added pumpkins, and then set up some cool posts. At this point the ambience of the parks was just perfect for that spooky Halloween vibe, with the purple lighting and bare branch silhouetted trees.

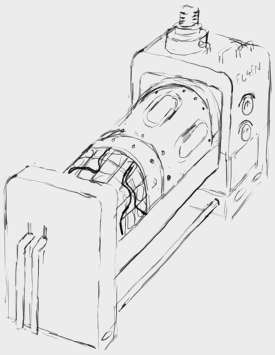
BUILDINGS

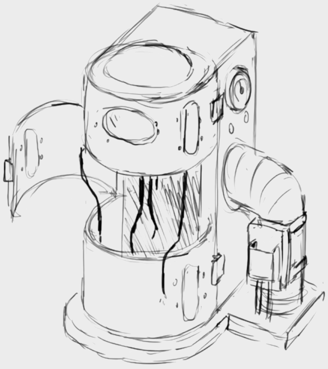
Sync Clinic

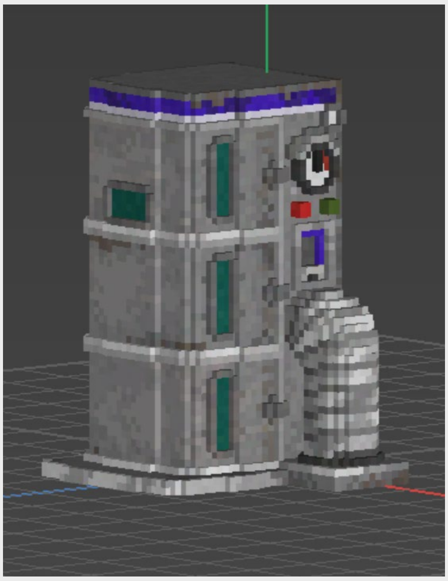
The sync machine is inspired by an iron lung machine, and other similar sci-fi style chamber designs. Most of the designs have a person lying down flat, however, I quite liked the upright Iron-Man/Captain America style chamber. Cole also pointed out that it was more akin to something one would find at a walk-in centre. The multi-door aesthetic added some cinematic quality to when it opens as well as if it could target specific body parts for cybernetic enhancements; legs, arms, body, etc.

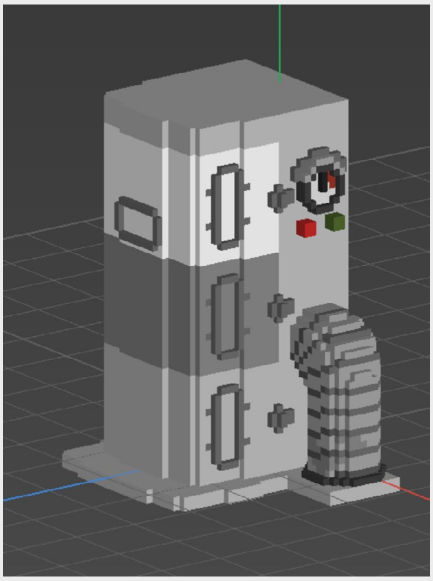


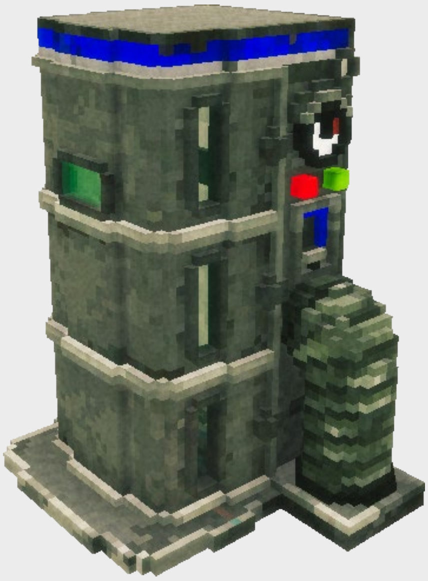












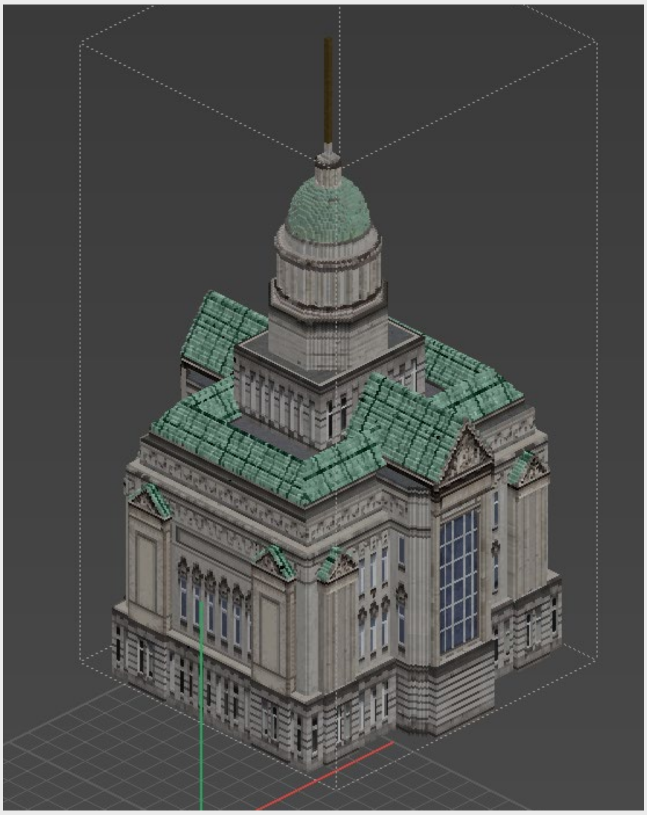
Many of the props used in the sync clinic were also used in the hospital. The medical props are all blue to show branding ownership of Elliot Genetics.

City Hall

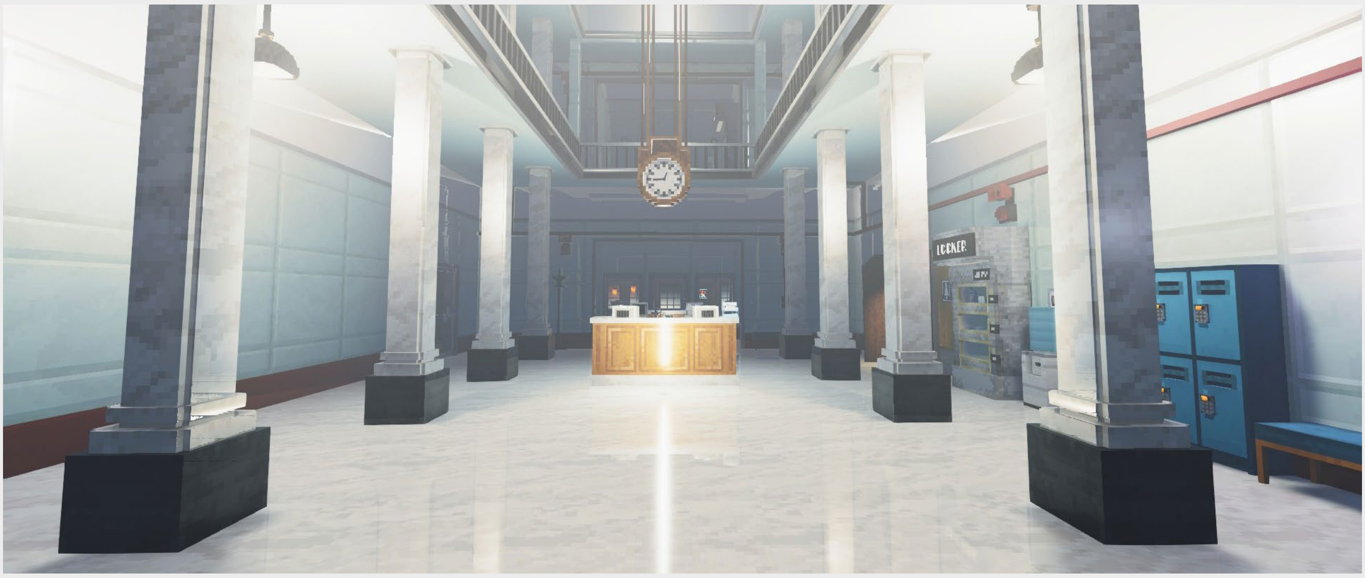
The city hall was a new addition to the city, and prime real estate for new voxel models.

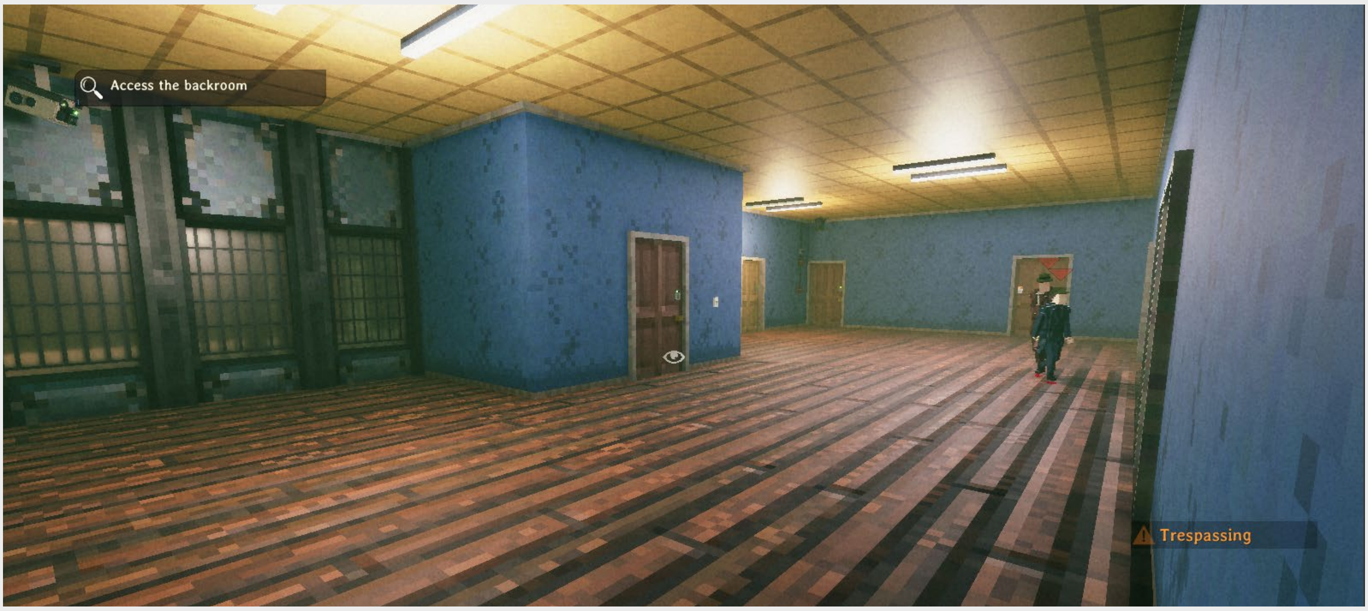
This introduced all the new marble assets which were inspired by the grand central station and other various grand city hall lobbies.

The addition of the shiny marble floors was perfect for interiors to display high levels of wealth. And it inspired my to then to create the marble walls and pillars to complete the look for the city hall.



A small model of the city hall, something for citizens interested in architecture to put on their shelf!

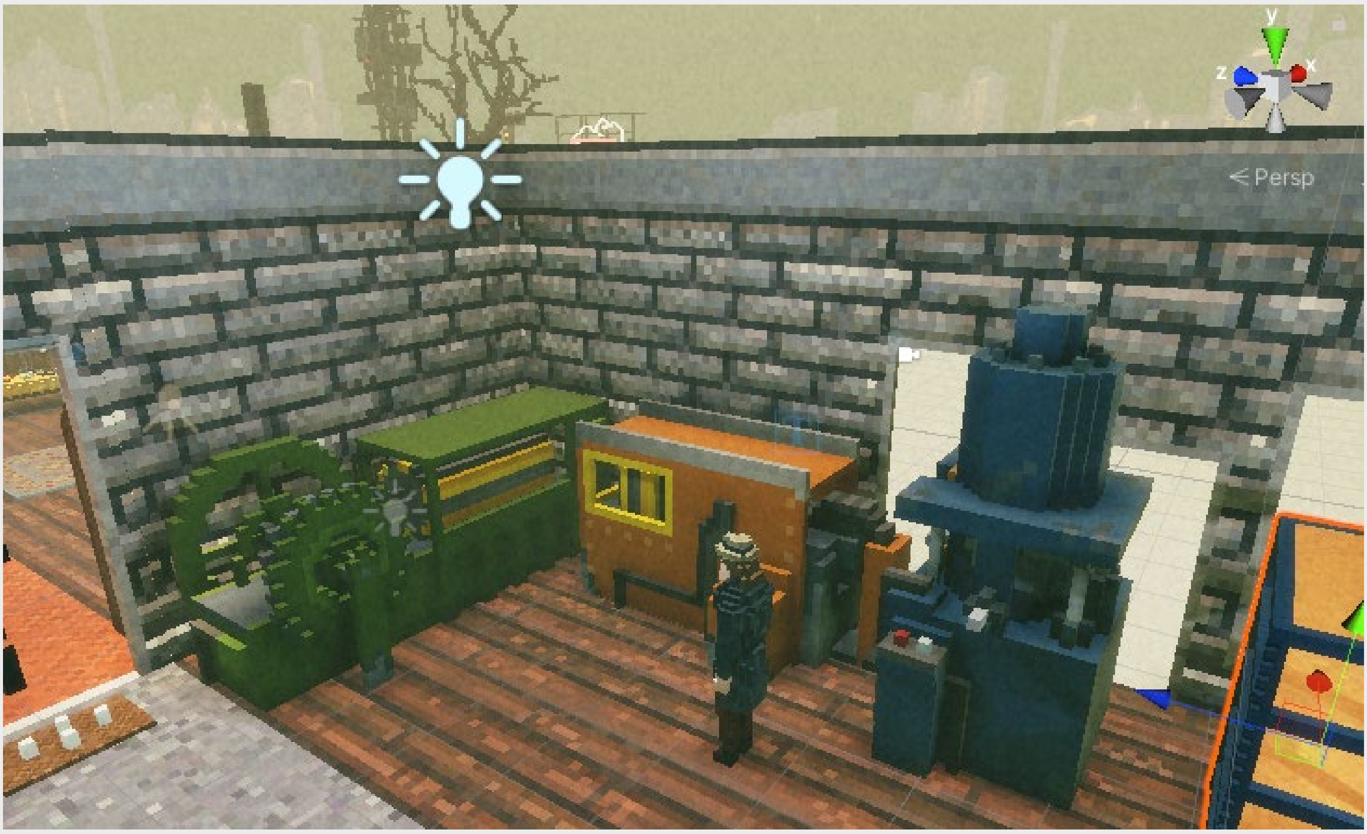


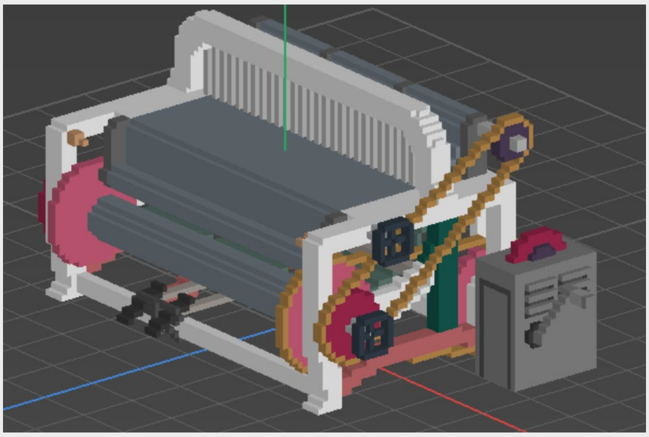


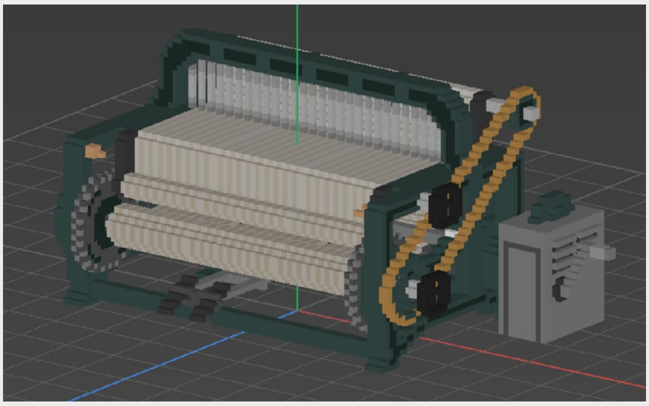
An old version of the factories, empty, liminal, waiting for industry.

Factory

Here's some machines for the first design of the factory. However, when the floors were remade at a higher resolution so too were the factory machines. This is what led to the creation of the current machines.

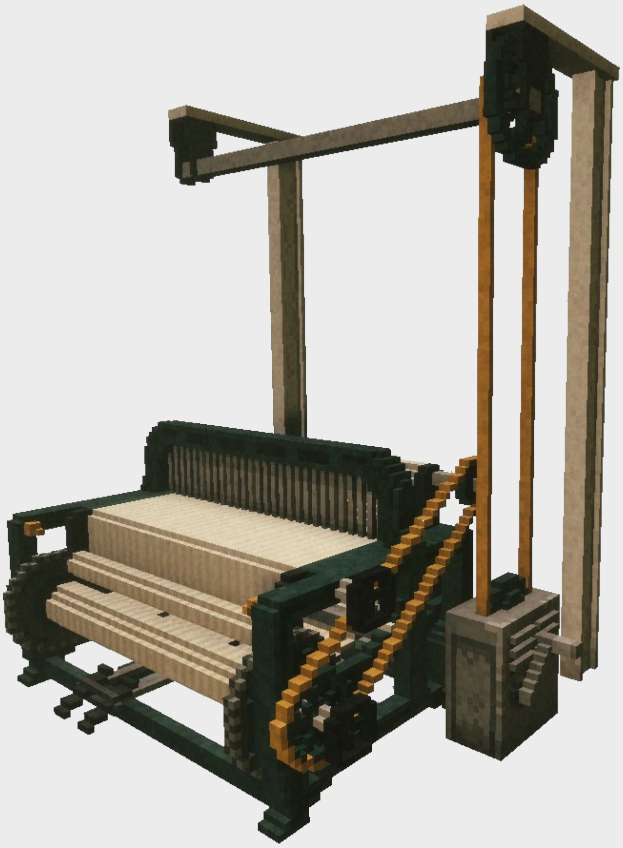


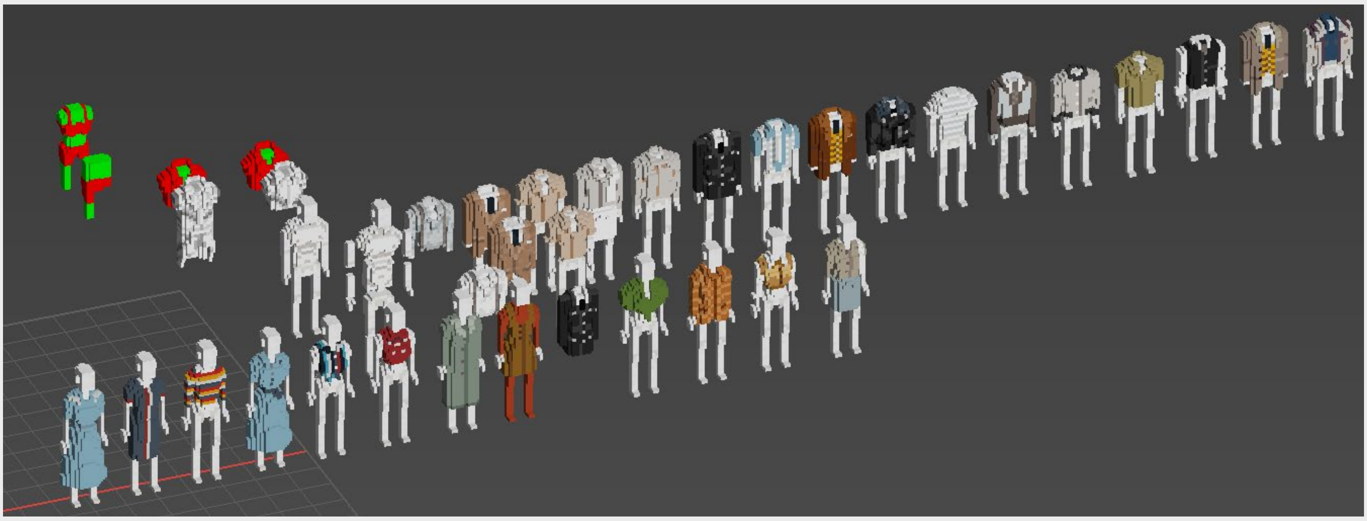




In 1945 Toyota took over the Kariga-Minami plant, a former spinning plant facility, and repaired 40 of 611 surviving looms. The images of this mechanical industrialisation inspired me to create the loom model for the factories.

For the brick texture of the walls I reduced the number of colours in a brick photograph texture and then downscaled it. After this I ensured it tiled both vertically and horizontally, as I tidied up any artefacts left over from the first few steps. This included reducing the noisiness of the top mortar section, and redrawing any bricks at the edges of the texture, as well as tidying up the lines of mortar across the entire brickwork.





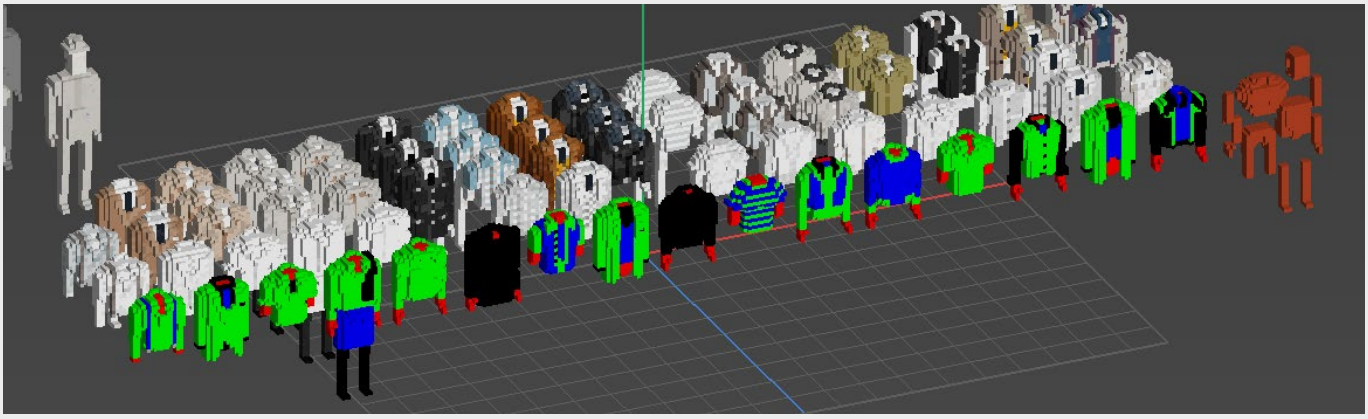
CHARACTERS

November 2020 marked the moment when the characters would be worked on.

In Cole's initial prototype all characters would wear trench coats. While appropriately noir, it lacked the expression of individuality that this simulator needed. Having a diversity of outfits created some individuality to help bring the game towards reality.

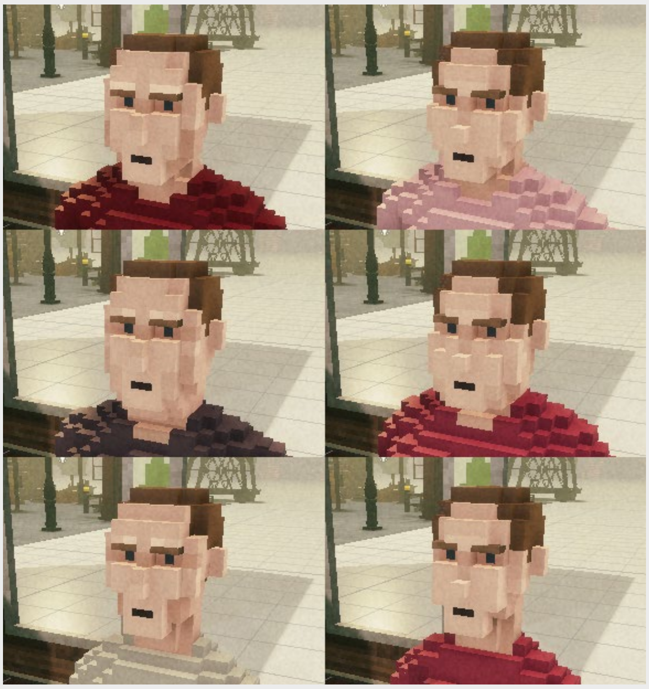
I started out by researching as many different vintage outfits as I could. Finding things that would fit within the era for all kinds of different situations - fancy, everyday, etc. and then there were work specific outfits I concepted too. When I first touched upon the outfits, the citizens were still a lower resolution. I made loads of designs, but then had to upscale them later.

I would create what I wanted the outfit to look like as all one solid model and then they were cut up into specific body parts - upper torso, lower torso, upper arms, lower arms, hands, etc... Overlapping parts of limbs were rebuilt, specifically those in between the joints e.g. the elbows. This was a time consuming task, and also as I described 'nauseating', rotating the model, switching between parts, exporting, importing, recreating it for different body shapes and trying to keep it consistent, and particularly when you had to factor in the number of different texture maps each model needed. Certain outfits also needed extra fabric layers, to reduce clipping, such as the jacket needed to go down over the legs but also needed to move with the legs.



However, to then maintain the clean shape of a blazer jacket as the citizens were animated walking was a difficult challenge.

Many iterations were needed for the clothing to remove clipping. As outfits were intended to be procedurally generated, different tops were tested with different bottoms and tweaked back and forth between the Unity editor and Qubicle. This was again a time consuming process that often felt very opaque to work with, and required a lot of trial and error.



I wrote a blog in 2021 that details a lot of the process:

July 30th, 2021

Creating faces in low resolution

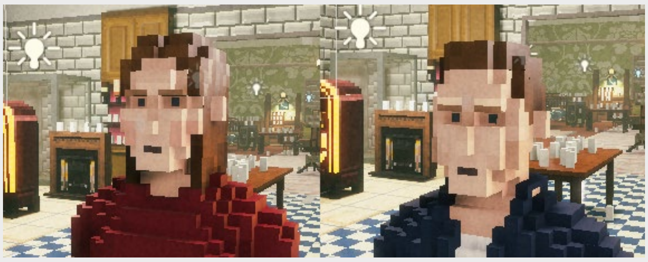
Recently we have been giving the citizens of Shadows of Doubt an overhaul. We have upped their resolution slightly, given them new clothes, and most importantly faces, with eyes and mouths and ears! We are also working on the citizen's style choices: beards, hairdos, moustaches, tops, hats, dresses, etc. All of this should give a bit more life to the people walking around the streets of the city as well as a recognisable and individual identity.

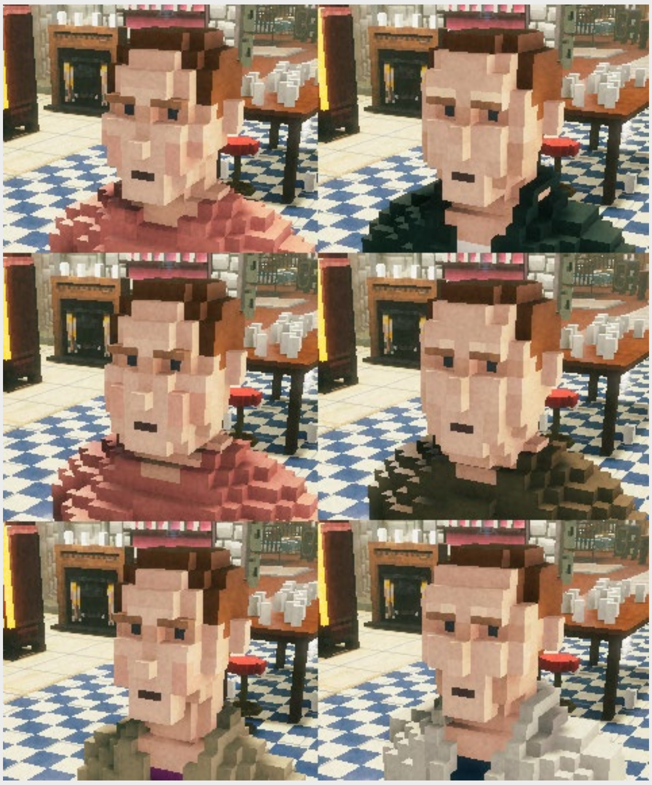
One of the biggest challenges with voxel faces is adding subtle differences in order to generate diversity. It is particularly difficult to make masculine and feminine faces distinct and believable at such a low resolution. There are some key features that help communicate whether a face is masculine or feminine; primary details such as, head shape, brow line form, chin, cheek and jawline structure, hairline position, nose shape, lips. Then there are secondary details like hairstyle and facial hair which can be used for further emphasis.

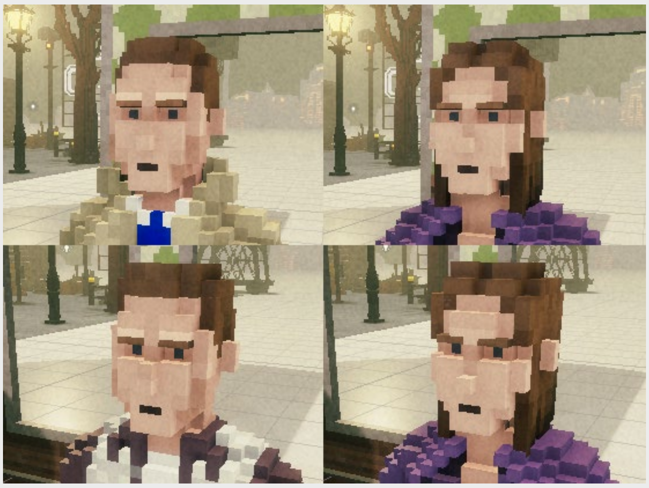
[To the left] is a comparison between the masculine and feminine heads and their different face shapes. I will explain the design choices I made, and hopefully, some of you can benefit from this analysis to help create your own voxel characters, or you might simply find it interesting.

Primary details

One of the most notable details is the overall head shape, more rounded for feminine heads and squarer for masculine heads. To portray roundness there is an extra voxel between the top of the forehead and the







hairline, softening the interface between the face and the top of the head. For masculine faces the hairline tends to start higher up, therefore by having the forehead 3 voxels tall the illusion that the hairline starts higher up is created. In addition to this, male pattern baldness can be created by simply having the voxels of the corner of the forehead be skin coloured, which also makes the heads appear squarer.

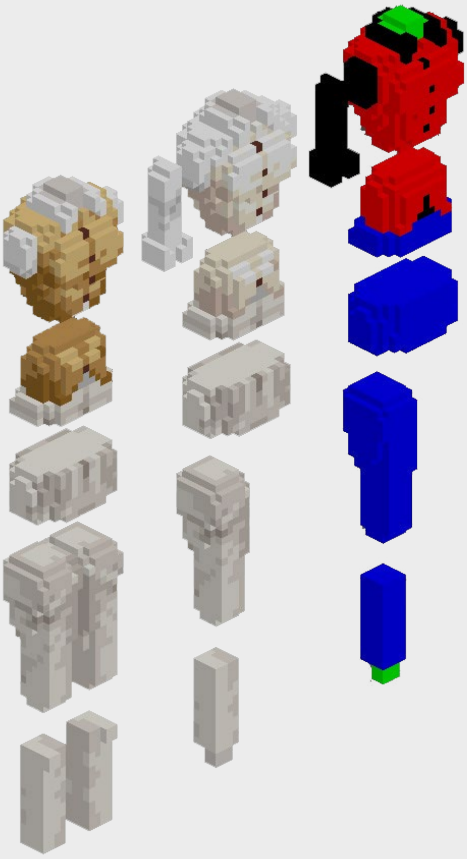
A wider brow is a notable masculine trait, this is simply done by making the brow 5 voxels wide. Prominent cheekbones are a more feminine characteristic, however, on both styles, the face width is 5 voxels wide. Yet, by having the brow of feminine heads only 3 voxels, the cheekbones appear to sit further out and thus appear more prominent.

Another masculine feature is that of the nose, the bridge of the nose starts higher up, and this is simply communicated with having two voxels for the nose. Whereas, for a feminine face the nose is only one voxel. The added benefit of this is that it helps emphasise the roundedness or squareness of the head.

Masculine faces typically have stronger, wider chins when compared with feminine faces, however, this detail is difficult to communicate without trying to make feminine chins be only a single voxel wide or underdeveloped. Instead, simply a standard neutral ground for both chins was chosen.

Secondary details

Once these primary details are figured out, then hairstyles and facial hair can be added. Having a base hair area textured onto the head helps ensure the haircut models work seamlessly and look natural. Hair colour and skin colour can be randomised using the colour shaders and a whole plethora of unique looking citizens cans be created.



If you follow us on social media, you may have noticed that the citizens' wardrobe selection has expanded beyond trench coats and trilbies. Don't worry, you'll still find this dapper detective look around the city. However, you'll also see a range of other outfits; from casual jeans and shirts to snazzy dresses and to classy suits.

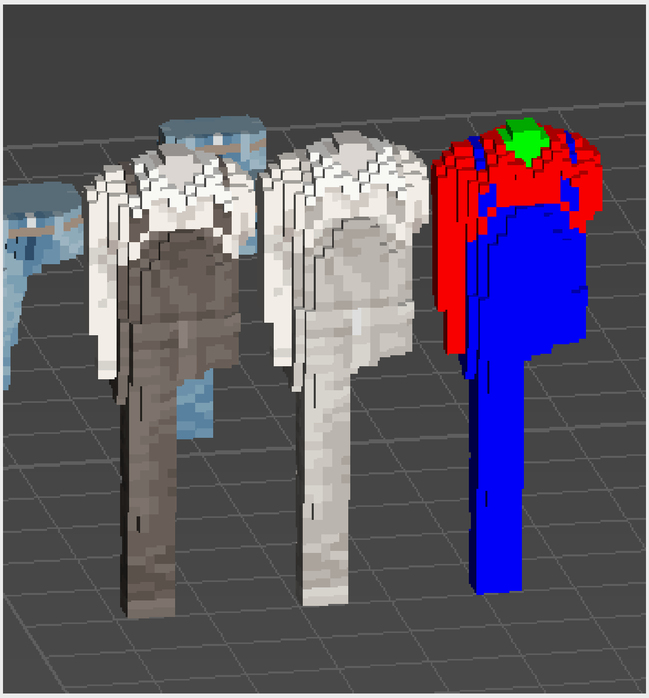
In this blog, I'll run you through what it takes to create a new outfit for the game and the decisions I make when designing them.

Creating the Outfits

All designs start with a little bit of research. Usually, this entails googling I920-80s clothing and finding some simple inspiration there. However, I'll often look up outfits worn by celebrities from this time, or I'll draw inspiration from noir films, such as 'The Maltese Falcon'.

To model an outfit there are a few important limitations that need to be addressed to get it working in Shadows of Doubts. For the purposes of procedural generation, every outfit, ideally, can be worn by every citizen and have a random colour. Therefore, when an outfit is modelled, it is modelled six times. Three times for each body type we have, and then another three times so that there is a masculine and feminine version.

To get the random colours, the model is first created fully coloured so I can get a good idea of how it "should" look. It is then manually grey-scaled to allow the code to recolour it properly, and with this, I create the colour map to select each area to be recoloured differently. Generally, the outfits have 3 colours to be replaced: red, green and blue, however, I'll also use black on the colour map to maintain certain colours from the base texture before being recoloured by the code, like a white shirt or black buttons.



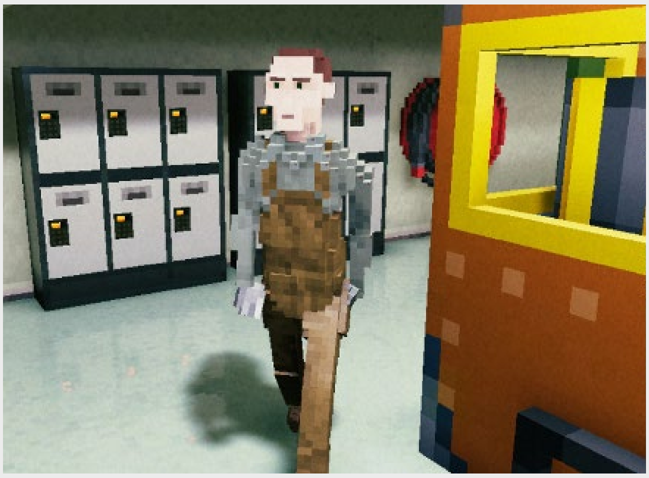
Once this first version is completed each model needs to be checked to make sure it works with the animation and other outfits that it'll randomly generate with. The outfits replace the models for the arms, legs, body, etc. so making sure that they all align properly and there are no oddities sticking out when a character is walking is important. A good amount of trial and error is needed, going back and forth between Unity and Qubicle, chopping off voxels here and there to ensure the outfit works well. Generally, the tops and the bottoms, as well as dresses, are built from a standardised base design and this helps reduce the amount of trial and error needed to rework the outfits to a good standard.

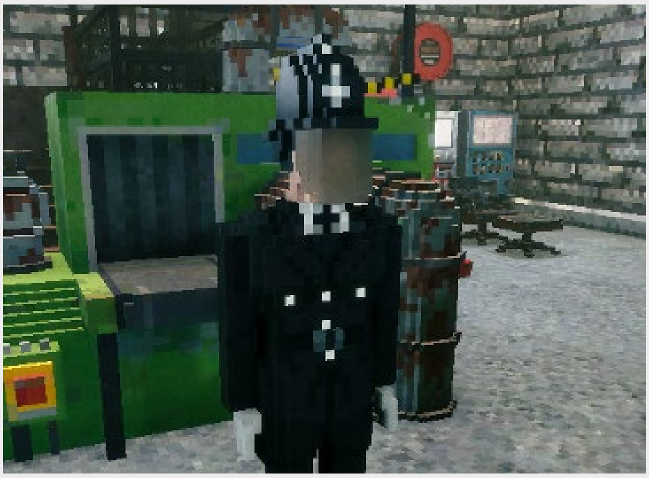
With voxel models, the texturing and the modelling is often one in the same process. Particularly for lower resolution voxel models. It's very easy to simply retexture while working in Qubicle. Moreover, the folds of clothing and the surface texturing is baked into the base texture model, rather than using polygons. This means combining the texturing and modelling processes is important because it'll help me make decisions as to how to get a good clothing "look".

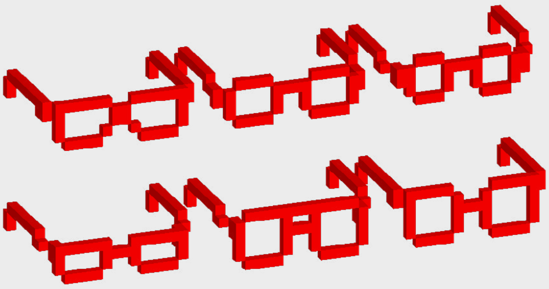
Unique Outfits

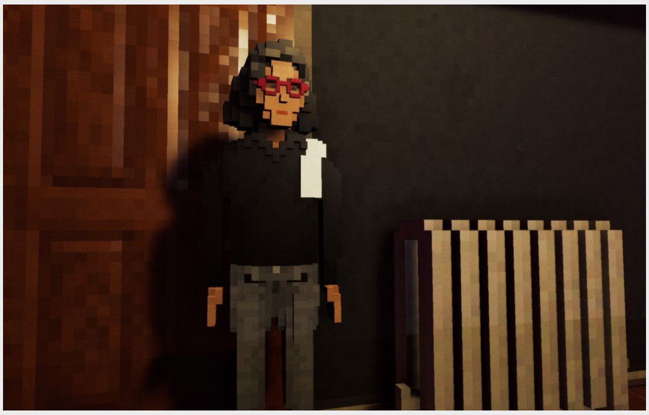
While it is desirable to make all tops and bottoms work together to get a huge diversity of procedurally generated outfits, there are some cases where this simply wouldn't allow for enough shape variation. The biggest cause of this is that the tops and bottoms need to sit flush at the interface of where they meet at the waist. If this interface differs between each outfit there's usually some weird clipping when you start to mix and match them.

To get around this there are a few "complete" outfits, these are ones that cover the full body and don't mix and match several pieces.









It's far easier to create a unique shape if the whole outfit works as one. The gold jacket and white trousers in the screenshot above is one such example. Another example is the dresses. Which also provided another significant challenge…

There are no custom animations for the outfits, therefore for dresses, the design must allow for leg movement without the model deforming to keep the fixed voxel shape. As a result, the dresses are made as a solid objects because this means as the legs split apart for the walking animation they appear as one continuous surface. If they were hollow, you would essentially see the dress split apart too when walking.

A similar design is used for the trench coats, but, as these maintain the models for the upper legs a weird little wall of voxels is used to hide when the legs split apart when walking. However, this may need further work if it poses an issue during sitting animations where you can see the underside of the upper leg models.

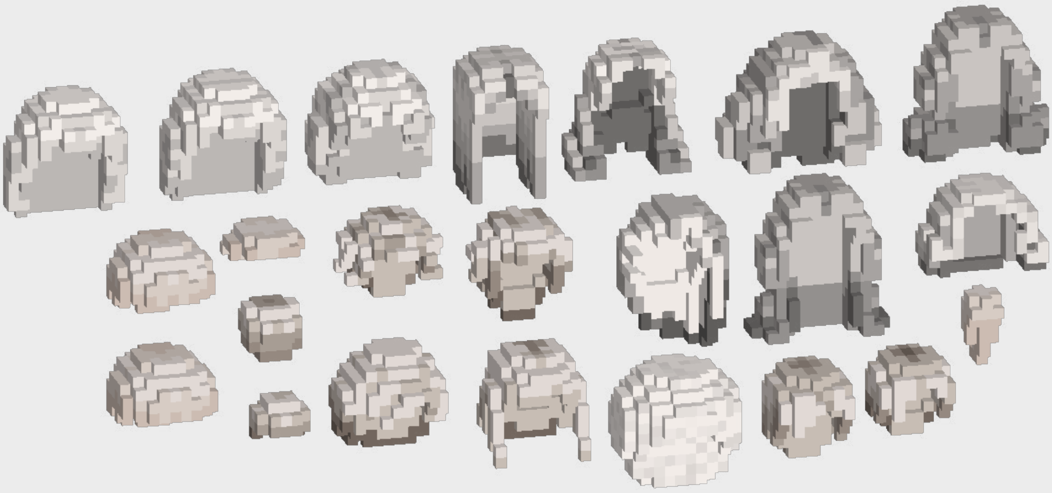
Other notable unique outfits are those that are used for specific professions. For factory workers, I've created some lovely work overalls to wear instead of getting their trench coats dirty! And then for the police officers, we created a somewhat dystopian twist on the traditional British police uniform. They've got the vintage smart black uniform, and custodian helmet, however, we've attached a riot gear visor. This helps enforce the idea that they're privatised security and gives them an impersonal feel as it obscures their identity.

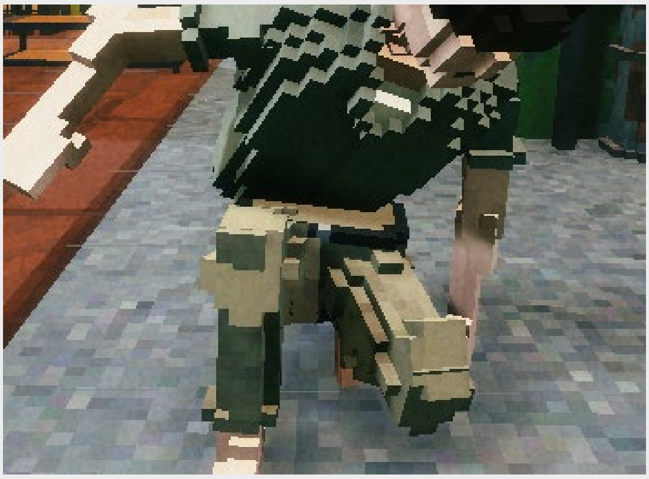
Other Features

Recently we introduced glasses and hats into the character generation - I find they really help add a great amount of personality to the characters, with very little effort.

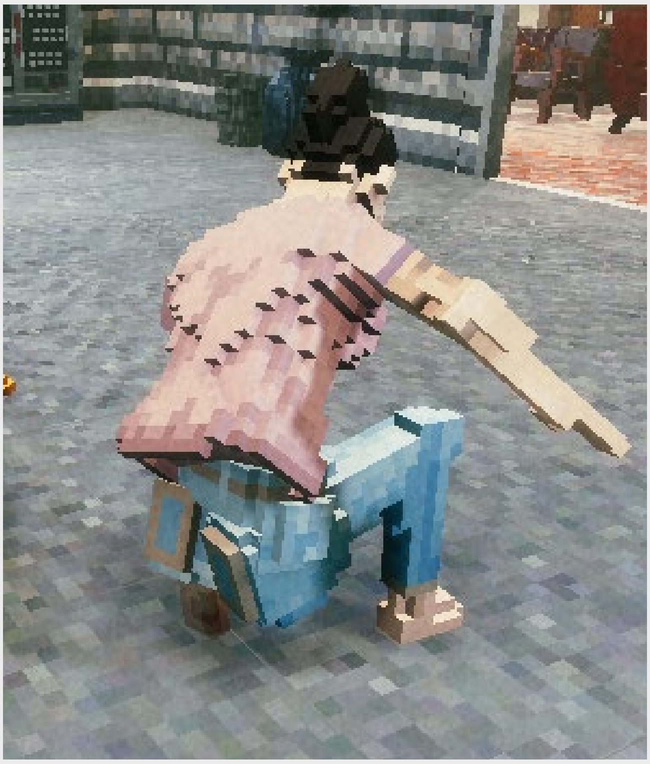
We also have a huge variety of hairstyles. Conveniently the hairstyles are designed to work with any head shape as they all follow the same structure, unlike the clothing. This means any single hairstyle can be used for all 6 head types.

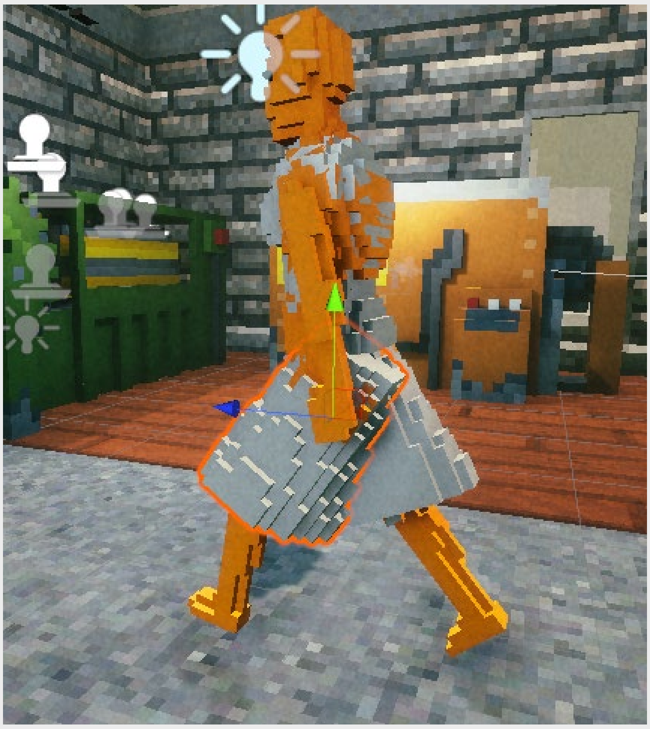
Lastly, shoes! The shoes aren't part of the outfits as such, because they can be an important clue for your detective work. Currently, there are 3 types of shoes, your bog-standard kind, boots and heels. They have a higher resolution to the rest of the citizens' outfits to allow you to distinguish the detail on them.





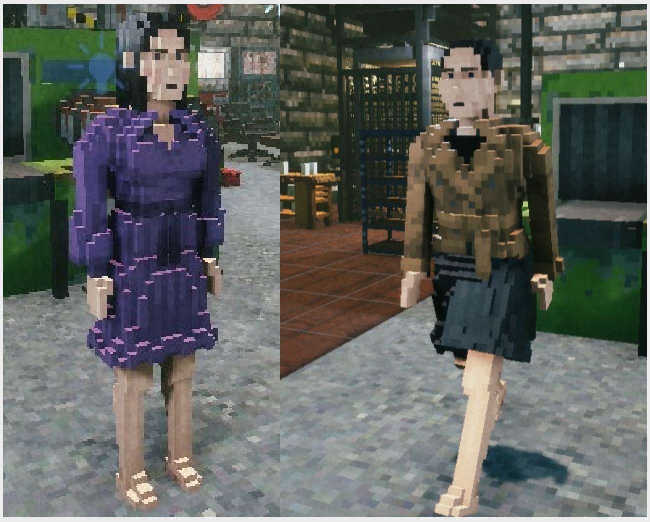
Here's a particularly egregious instance where the model simply doesn't work with an animation. Whilst it looks fine in a standing position, once crouched the top texture of the trousers used for the belt, now becomes part of the characters butt and looks like it's got a couple of picture frames attached. Another issue here is that the midriff of the character needed to be split in half so that the animation could bend in the middle of the body, this meant going back through a tonne of models and cutting the torso in half.





Another difficult clothing task to tackle was the dresses. How to give a fabric look to solid objects, without looking stiff when walking...

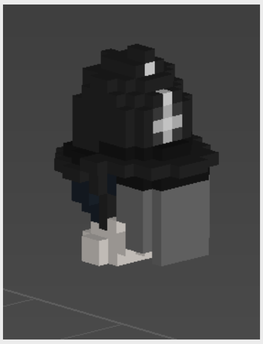
I really enjoyed creating the dresses for Shadows. However, the long trench coat I had already designed would be clipped through by the dresses. Instead what I did was create a short trench coat variety that ended above the legs, this also worked to create a more feminine style of coat.

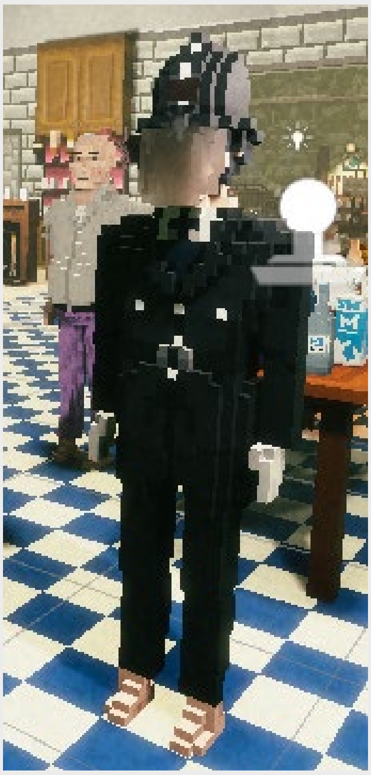


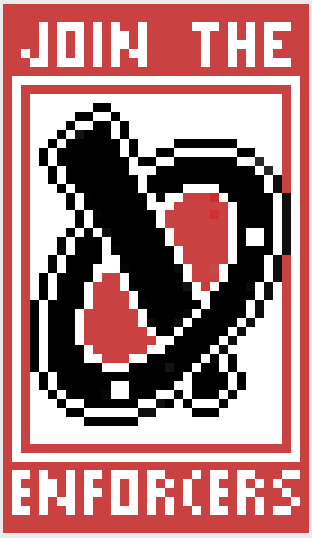
Police

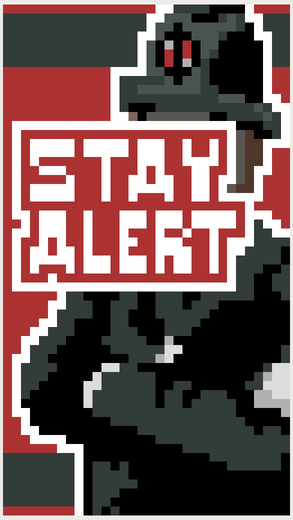
For the "police" we wanted something menacing, that conveys their corrupt privatisation. Really they're glorified security guards, given more jurisdiction than they morally should have. However, in this world, it is the corporations which hold the power. Really they act more as security guards than police. Using a riot face mask helped emphasise the heavy handed attitude of the corporations. As well as removing their individuality.

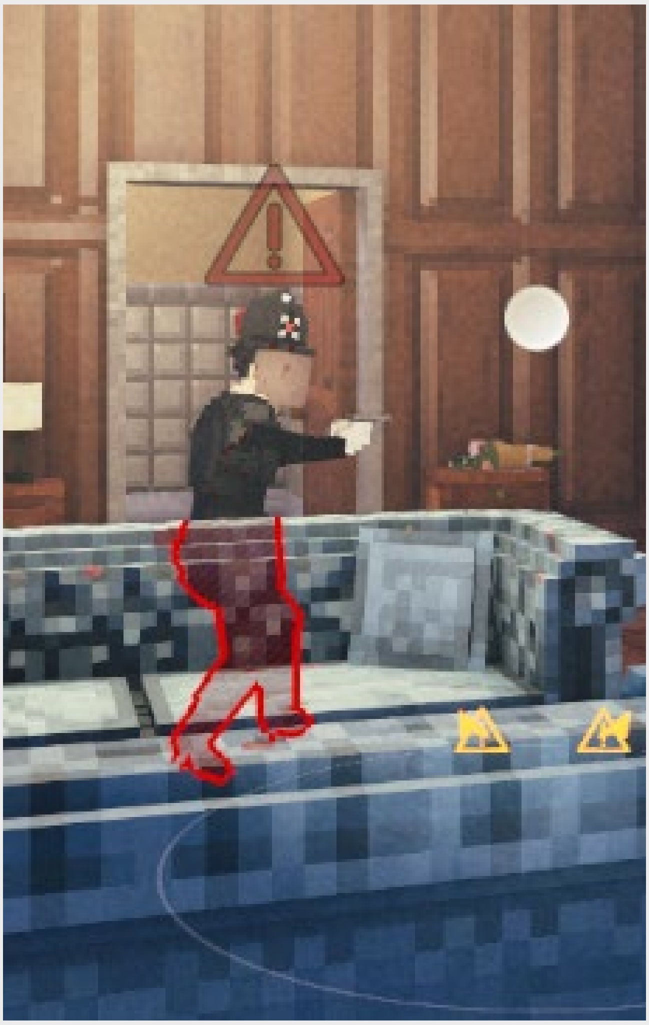
Red lines and the security logo emblem was added to their shoulders. This was to indicate the ownership of the police by Starch, as well as make them feel like Nazis...

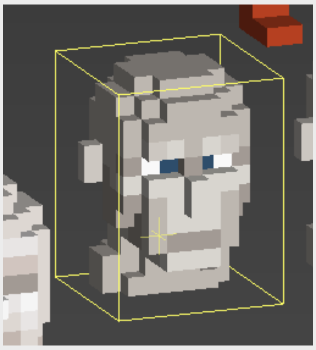


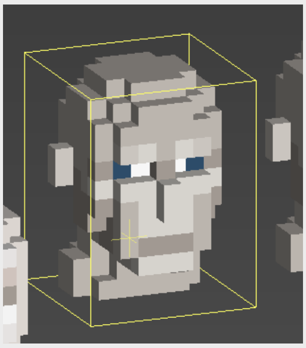


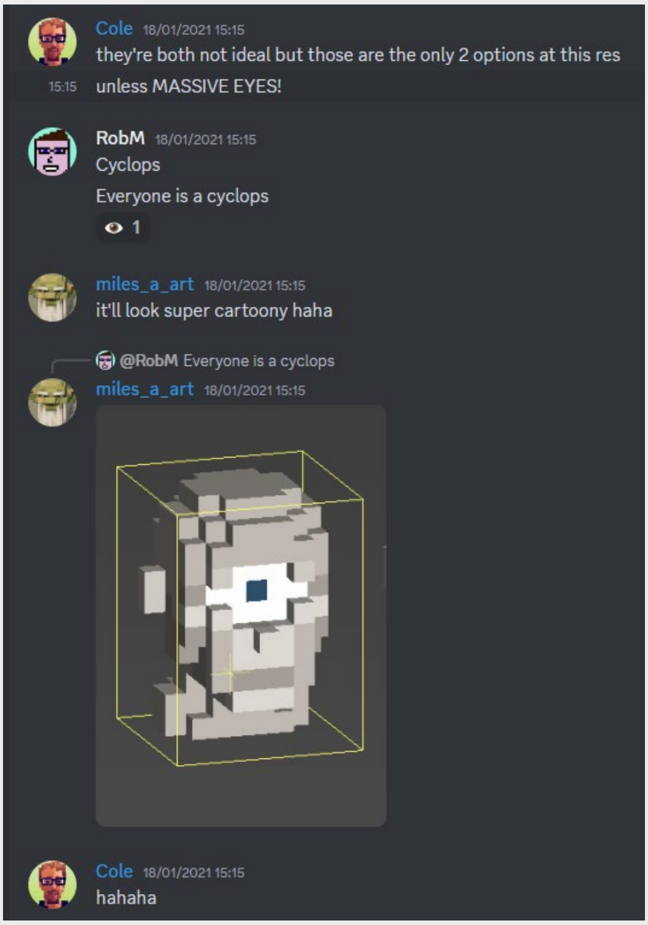












Cole 18/01/2021 15:15
they're both not ideal but those are the only 2 options at this res
15:15 unless MASSIVE EYES!

RobM 18/01/2021 15:15
Cyclops
Everyone is a cyclops
👁 1

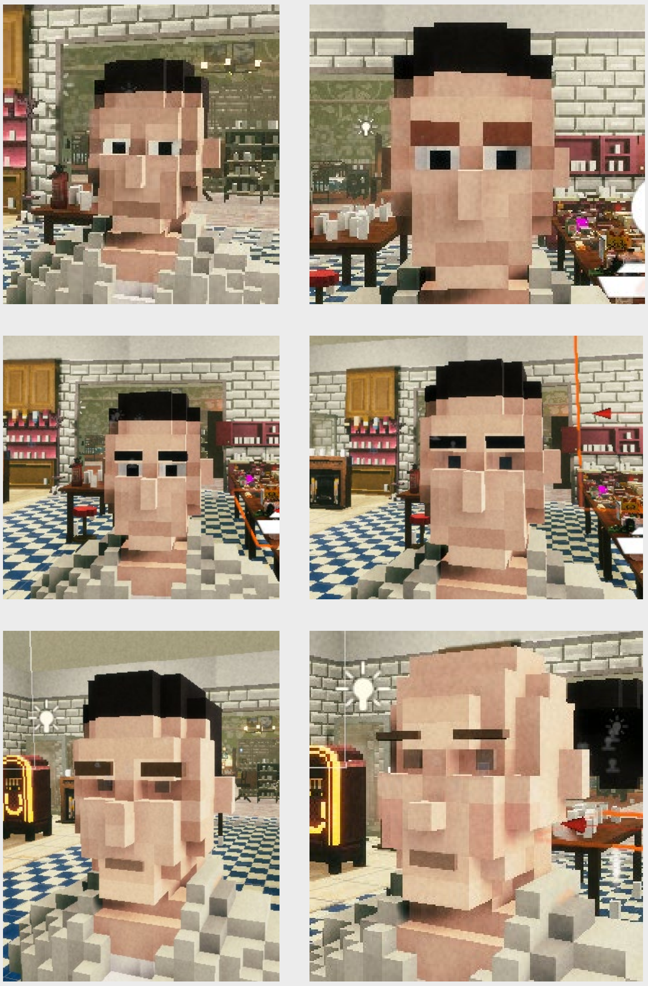
miles_a_art 18/01/2021 15:15
it'll look super cartoony haha

@RobM Everyone is a cyclops
miles_a_art 18/01/2021 15:15

Cole 18/01/2021 15:15
hahaha

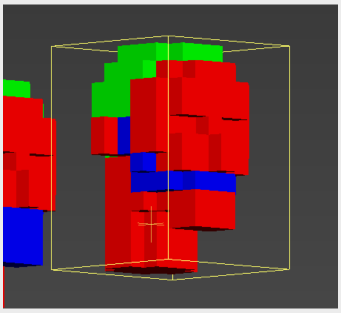
Faces

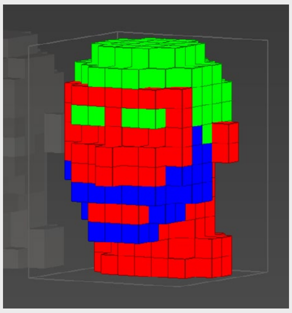
Now that the outfits were making good progress the next aspect was to fix the faces. For the initial low resolution models, the heads were simple cubes with hats and hair and ears. When the resolution was increased they became eyeless horrors. The question was then, how to give them eyes without landing in an uncanny valley or the cross-eyed beauties of Minecraft.



The face went through several iterations of designs. At first we tried having the mouth, the eyes and the eyebrows baked into the texture of the head. However, the scale of these relative to the rest of the head was off. Instead bit by bit they were split off and turned into separate objects. This was more in line with the gritty realism of the game, and it also enabled animated expressions.

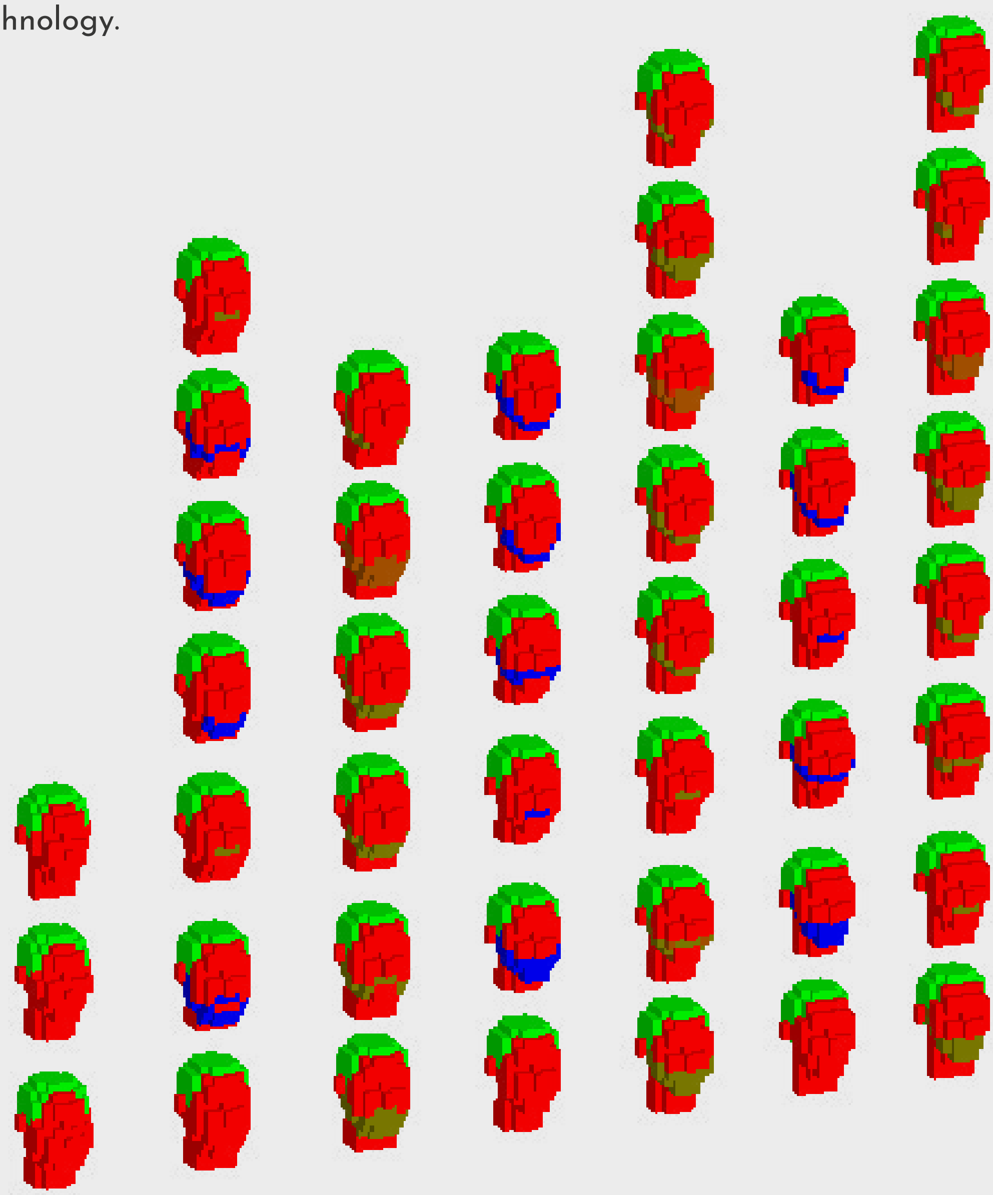
Facial Hair





Having explored colour maps with the clothing I realised they could also be used for organic elements like the hair. This would allow facial hair and hair cuts to sit more seamlessly on the heads so they didn't look "glued" on. Additionally this added many levels of face variety without any additional models, instead the texture maps could simply be changed.

The off green texture map colour is a mix between the green and red, what this creates is a sort of sparse stubble when the colour map is applied in Unity using the pre-existing shader technology.



To add to the cyberpunk element of the game, some cooler, tech-y haircuts were added.

Along with haircuts and facial hair, glasses were another addition to the character customisation that added significant variation. However, when added into the game at first, everyone was wearing glasses... it seemed mass myopia was plaguing the cities!

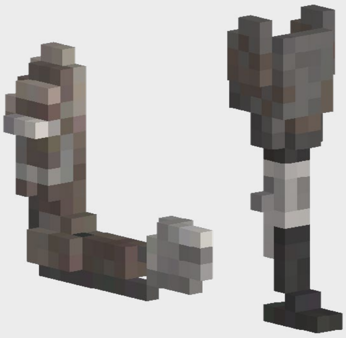
Many elements of creating the characters came as a result of an accident, or simply playing around with the models as I was adding them into Unity. Two hairstyles could be created, by simply moving this hair model 1 voxel forwards or backwards. This was helpful as it saved time and resources.

The cyberpunk haircuts worked nicely when combined with a more vintage outfit for something really fresh and unique for the visuals of the game.

I also love this image as this guy looks like a secret agent straight out of a film.

More and more cool outfits started appearing in the proc gen, like this perfect starch cola brand ambassador

Something we really wanted to add was prosthetics. As Shadows is a cyberpunk-style game, it only made sense to make them as if they were cybernetic enhancements. However, to fit in with the pre-existing technology of the game they had to be low end tech, more like simple robotics, or maybe even without electricity, just all mechanical.



This wraps up our exploration into the voxel model development process. I hope you've enjoyed taking a peek behind the development curtain.

Final Thoughts...

That about wraps up the journey for the visuals of Shadows: The game didn't change much visually during the last few years of development, instead the focus was on completing the rest of the game. Something that took a good while due to the complexity of the interlocking systems. The disjointed nature of the initial development phase (as you read about early on in this very book), lead to some rather rough parts of the codebase. Something that cost a good amount of time to navigate later in the project.

It's easy for me to forget just how long a journey this project is; a project that has spanned 8 years of my life and counting. Shadows is not a perfect project; rough around many edges, but for what it is; an attempt at an ultra-ambitious dream project by a very small team on limited budget, I'm so pleased about what the team has achieved.

Shadows of Doubt wouldn't have been a success, or even existed, without the support of the publisher Fireshine. This project was never going to be easy, but the folks there have supported the development every step of the way. A big shout-out to them, some unsung heroes of the project.

At the time of writing this the journey isn't over; we'll be supporting the game post launch for a good while. But I shall carry forward a huge amount of learning and experience from Shadows of Doubt to whatever comes next...

SHADOWS OF DOUBT

‘Shipping on the Clyde’ John Atkinson Grimshaw, sourced from Pixabay
Book design by Code: E Design